Efficient Dense Scene Flow from Sparse or Dense Stereo Data

Andreas Wedel[1,2], Clemens Rabe[1], Tobi Vaudrey[3],
Thomas Brox[4], Uwe Franke[1], and Daniel Cremers[2]

[1] Daimler Group Research {wedel,rabe,franke}@daimler.com
[2] University of Bonn dcremers@cs.uni-bonn.de
[3] University of Auckland t.vaudrey@auckland.ac.nz
[4] University of Dresden brox@inf.tu-dresden.de

Abstract. This paper presents a technique for estimating the three-dimensional velocity vector field that describes the motion of each visible scene point (scene flow). The technique presented uses two consecutive image pairs from a stereo sequence. The main contribution is to decouple the position and velocity estimation steps, and to estimate dense velocities using a variational approach. We enforce the scene flow to yield consistent displacement vectors in the left and right images. The decoupling strategy has two main advantages: Firstly, we are independent in choosing a disparity estimation technique, which can yield either sparse or dense correspondences, and secondly, we can achieve frame rates of 5 fps on standard consumer hardware. The approach provides dense velocity estimates with accurate results at distances up to 50 meters.

1 Introduction

A very important feature to extract from a moving scene is the velocity of visible objects. In the scope of the human nerve system such perception of motion is referred to as kinaesthesia. The motion in 3D space is called *scene flow* and can be described by a three-dimensional velocity field.

With images from a single camera, scene flow computation is clearly underdetermined, due to the projection on to the image plane. Even if the camera is moving, there arise ambiguities between the camera motion and the motion

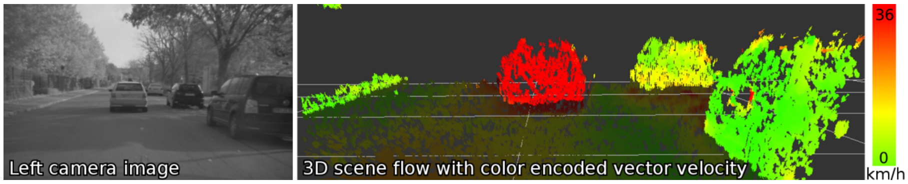

Fig. 1. Scene flow example. Despite similar distance from the viewer, the moving car (red) can be clearly distinguished from the parked vehicles (green).

of objects in the scene. These ambiguities are largely resolved when using a second camera. Ambiguities only remain due to missing structure in local parts of the image. In 2D motion estimation, this is known as the aperture problem. A common way to deal with this problem is by using a variational framework (e.g. [6]), which includes a smoothness assumption on the velocity field. This allows for dense motion estimation, despite missing structure in parts of the image.

In the case of 3D motion estimation, we can also make use of a variational technique in order to achieve dense estimates (as done in [7]). However, it should be clear that only the motion of visible parts of the scene can be estimated. For our purposes, we refer to dense scene flow as the 3D velocity vector at each 3D point that can be seen by both cameras.

Scene flow estimation, with known camera parameters, involves estimating the 2D velocity in consecutive stereo frames, and also the disparity needed to calculate the absolute position of the world point. In this paper, we suggest performing the velocity estimation and the disparity estimation separately, while still ensuring consistency of all involved frames. The decoupling of depth (3D position) and motion (3D scene flow) estimation implies that we do not enforce depth consistency between t and t+1. While splitting the problem into two sub-problems might look unfavourable at a first glance, it only affects the accuracy of the disparity estimate and has two important advantages.

Firstly, the challenges in motion estimation and disparity estimation are very different. With disparity estimation, thanks to the epipolar constraint, only a scalar field needs to be estimated. This enables the use of efficient global optimisation methods, such as dynamic programming or graph-cuts, to establish point correspondences. Optical flow estimation, on the other hand, requires the estimation of a vector field, which rules out such global optimisation strategies. Additionally, motion vectors are usually much smaller in magnitude than disparities. With optical flow, occlusion handling is less important than the sub-pixel accuracy provided by variational methods. Splitting scene flow computation into the estimation sub-problems, disparity and optical flow, allows us to choose the optimal technique for each task.

Secondly, the two sub-problems can be solved more efficiently than the joint problem. This allows for real-time computation of scene flow, with a frame rate of 5 fps on QVGA images (320 × 240 pixel). This is about 500 times faster compared to the recent technique for joint scene flow computation in [7]. Nevertheless, we achieve accuracy that is at least as good as the joint estimation method.

1.1 Related Work

2D motion vectors are obtained by optical flow estimation techniques. There are dense as well as sparse techniques. Sparse optical flow techniques, such as KLT tracking [16], usually perform some kind of feature tracking and are preferred in time-critical applications, due to computational benefits. Dense optical flow is mostly provided by variational models based on the method of Horn and Schunck [6]. Local variational optimisation is used to minimise an energy function that assumes constant pixel intensities and a smooth flow field. The basic framework of Horn and Schunck has been improved over time to cope with discontinuities in the flow field, and to obtain robust solutions with the presence of outliers in image intensities [9]. Furthermore, larger displacements can be estimated thanks to image warping and non-linearised model equations [1, 9]. Currently, variational techniques yield the most accurate optical flow in the literature. Real-time methods have been proposed in [2, 18].

Scene flow involves an additional disparity estimation problem, as well as the task to estimate the change of disparity over time. The work in [11] introduced scene flow as a joint motion and disparity estimation method. The succeeding works in [7, 10, 19] presented energy minimisation frameworks including regularisation constraints to provide dense scene flow. Other dense scene flow algorithms have been presented in multiple camera set-ups [13, 17]. However, these allow for non-consistent flow fields in single image pairs. At this point it is worth noting that, although we separate the problems of disparity estimation and motion estimation, the method still involves a coupling of these two tasks, as the optical flow is enforced to be consistent with the computed disparities.

None of the above approaches run in real-time, giving best performances in the scale of minutes. Real-time scene flow algorithms, such as the one presented in [14], provide only sparse results both for the disparity and the velocity estimates. The work in [8] presents a probabilistic scene flow algorithm with computation times in the range of seconds, but yielding only integer pixel-accurate results. In contrast, the method we present in this paper provides sub-pixel accurate scene flow in real-time for reasonable image sizes. Furthermore, in combination with a dense disparity map the scene flow field is dense.

2 Scene Flow and its Constraints on Image Motion

2.1 Stereo Computation

Assume we are given a pair of stereo images. Normal stereo epipolar geometry is assumed, such that pixel rows y for the left and right images coincide. In practice, this is achieved by a rectification step, which warps the images according to the known intrinsic and relative extrinsic configuration of the two involved cameras [4, 12]. In addition, the principal points of both images are rearranged, such that they lie on the same image coordinates (x_0, y_0). A world point (X, Y, Z) given in the camera coordinate system is projected onto the image point (x, y) in the left image, and the image point $(x + d, y)$ in the right image according to

$$\begin{pmatrix} x \\ y \\ d \end{pmatrix} = \frac{1}{Z} \begin{pmatrix} X f_x \\ Y f_y \\ -b f_x \end{pmatrix} + \begin{pmatrix} x_0 \\ y_0 \\ 0 \end{pmatrix} \tag{1}$$

with the focal lengths f_x and f_y, in pixels, and the distance b as baseline, in metres, between the two camera projection centres. The disparity value d therefore encodes the difference in the x-coordinate of an image correspondence. With known camera parameters, the position of a world point can easily be recovered from an (x, y, d) measurement according to Equation (1).

The goal of any stereo algorithm is to determine the disparity d, in order to reconstruct the 3D scene. This is accomplished by either matching a small window from the the left image to an area in the right image, or by calculating a globally consistent solution, using energy minimisation techniques. The issue with the presented scene flow framework is that we can employ any stereo algorithm. In Section 5 this is demonstrated, as we show results with various sparse and dense stereo algorithms.

2.2 Constraints on Image Motion

Assume we are given two consecutive pairs of stereo images at time t and $t+1$. Analogous to the optical flow field, the scene flow field is the projection of the three-dimensional motion field. It provides for each pixel a change in the image space (u, v, d') between the two stereo image pairs, where u and v is the change in image x and y respectively, and d' is the change in disparity, all in pixels. The three-dimensional velocity field can be reconstructed, if both the image measurement (x, y, d) and its change (u, v, d') are known. Leaving the estimation of d to an arbitrary stereo algorithm, we will now derive the constraints for estimating (u, v, d').

Let $I(x, y, t)^l, I(x, y, t)^r$ be the intensity value of the left and right image, respectively, at time t and image position (x, y). Using Equation (1), a correspondence between the left and right stereo image at time t can be represented as $I(x, y, t)^l$ and $I(x+d, y, t)^r$. Since the flow in y-direction has to be equal in both images due to rectification, the constraints for the optical flow in the left and right images are:

$$I(x, y, t)^l = I(x+u, y+v, t+1)^l \tag{2}$$

$$I(x+d, y, t)^r = I(x+d+d'+u, y+v, t+1)^r \tag{3}$$

If the disparity d is known, the right image at time t is redundant for solving the scene flow problem, because $I(x, y, t)^l = I(x+d, y, t)^r$. In practice, $I(x, y, t)^l = I(x+d, y, t)^r$ does not hold exactly even for perfect d, since we have illumination changes between two different cameras. Therefore, we use the optical flow constraints for the left and right camera images separately, as stated in the above formulas.

Calculating optical flow in the left and right image separately, we could derive the disparity change $d' = u^r - u^l$, where u^r and u^l denote the estimated flow fields in the left and right image, respectively. However, we introduce an additional constraint, enforcing consistency of the left and right image at $t+1$:

$$I(x+u, y+v, t+1)^l - I(x+d+d'+u, y+v, t+1)^r = 0 \tag{4}$$

3 A Variational Framework for Scene Flow

Scene flow estimates according to the constraints formulated in Section 2 can be computed in a variational framework by minimising an energy functional consisting of a constraint deviation term and a smoothness term that enforces smooth and dense scene flow:

$$E(u,v,d') = E_{\text{Data}}(u,v,d') + E_{\text{Smooth}}(u,v,d') \tag{5}$$

Integrating the constraints from Section 2 over the image domain Ω, we obtain the following data term:

$$\begin{aligned} E_{\text{Data}} = & \int_{\Omega} \Psi\Big(\Big(I(x+u,y+v,t+1)^l - I(x,y,t)^l\Big)^2\Big)\,dxdy \\ + & \int_{\Omega} c(x,y)\,\Psi\Big(\Big(I(x_d+d'+u,y+v,t+1)^r - I(x_d,y,t)^r\Big)^2\Big)\,dxdy \\ + & \int_{\Omega} c(x,y)\,\Psi\Big(\Big(I(x_d+d'+u,y+v,t+1)^r - I(x+u,y+v,t+1)^l\Big)^2\Big)\,dxdy \end{aligned} \tag{6}$$

where $\Psi(s^2) = \sqrt{s^2+\varepsilon}$ ($\varepsilon = 0.0001$) denotes a robust function that compensates for outliers [1], and $x_d := x + d$ for simpler notation. The indicator function $c(x,y) : \Omega \to \{0,1\}$ returns 0 if there is no disparity known at (x,y). This can be due to a point seen only in one camera (occlusion) or due to a non-dense stereo method. Otherwise $c(x,y)$ returns 1. The first term in Equation (6) imposes the brightness constancy for the left images, the second one for the right images, and the third one assures consistency of the estimated motion between left and right images.

The smoothness term penalises local deviations in the scene flow components and employs the same robust function as the data term in order to deal with discontinuities in the velocity field:

$$E_{\text{Smooth}} = \int_{\Omega} \Psi\left(\lambda|\nabla u|^2 + \lambda|\nabla v|^2 + \gamma|\nabla d'|^2\right)dxdy \tag{7}$$

where $\nabla = (\partial/\partial x, \partial/\partial y)$. The parameters λ and γ regulate the importance of the smoothness constraint, weighting for optic flow and disparity change respectively. Interestingly, due to the fill-in effect of the above regularizer, the proposed variational formulation provides dense scene flow estimates (u,v,d'), even if the disparity d is non-dense.

4 Minimisation of the Energy

For minimising the above energy we compute its Euler-Lagrange equations:

$$\begin{aligned} &\Psi'((I_z^l)^2)I_z^l I_x^l + c\Psi'((I_z^r)^2)I_z^r I_x^r + c\Psi'((I_z^d)^2)I_z^d I_x^r \\ &\quad -\lambda \operatorname{div}\left(\Psi'(\lambda|\nabla u|^2 + \lambda|\nabla v|^2 + \gamma|\nabla d'|^2)\nabla u\right) = 0 \end{aligned} \tag{8}$$

$$\begin{aligned} &\Psi'((I_z^l)^2)I_z^l I_y^l + c\Psi'((I_z^r)^2)I_z^r I_y^r + c\Psi'((I_z^d)^2)I_z^d I_y^r \\ &\quad -\lambda \operatorname{div}\left(\Psi'(\lambda|\nabla u|^2 + \lambda|\nabla v|^2 + \gamma|\nabla d'|^2)\nabla v\right) = 0 \end{aligned} \tag{9}$$

$$\begin{aligned} &c\Psi'((I_z^r)^2)I_z^r I_x^r + c\Psi'((I_z^d)^2)I_z^d I_x^r \\ &\quad -\gamma \operatorname{div}\left(\Psi'(\lambda|\nabla u|^2 + \lambda|\nabla v|^2 + \gamma|\nabla d'|^2)\nabla d'\right) = 0 \end{aligned} \tag{10}$$

where $\Psi'(s^2)$ denotes the derivative of Ψ with respect to s^2. We define $I_z^l := I(x+u, y+v, t+1)^l - I(x,y,t)^l$, $I_z^r := I(x_d+u, y+v, t+1)^r - I(x_d, y, t)^r$, and $I_z^d := I(x_d+u+d', y+v, t+1)^r - I(x+u, y+v, t+1)^l$, and subscripts x and y denote the respective partial derivatives of $I(x+u, y+v, t+1)^l$ and $I(x_d+d'+u, y+v, t+1)^r$.

These equations are non-linear in the unknowns, so we stick to the strategy of two nested fixed point iteration loops as suggested in [1]. This comes down to a warping scheme as also employed in [9]. The basic idea is to have an outer fixed point iteration loop that comprises linearisation of the I_z expressions. In each iteration, an increment of the unknowns is estimated and the second image is then warped according to the new estimate. The warping is combined with a coarse-to-fine strategy, where we work with down-sampled images that are successively refined with the number of iterations. Since we are interested in real-time estimates, we use only 4 scales with 2 outer fixed point iterations at each scale.

In the present case, we have the following linearisation, where k denotes the iteration index. We start the iterations with $(u^0, v^0, d'^0)^\top = (0,0,0)^\top$:

$$I(x+u^k+\delta u^k, y+v^k+\delta v^k, t+1)^l \approx I(x+u^k, y+v^k, t+1)^l + \delta u^k I_x^l + \delta v^k I_y^l \tag{11}$$

$$\begin{aligned} &I(x_d + d'^k + \delta d'^k + u^k + \delta u^k, y + v^k + \delta v^k, t+1)^r \\ &\quad \approx I(x_d + d'^k + u^k, y + v^k, t+1)^r + \delta u^k I_{x_d}^r + \delta d'^k I_{x_d}^r + \delta v^k I_y^r \end{aligned} \tag{12}$$

From these expressions we can derive linearised versions of I_z^*. The remaining non-linearity in the Euler-Lagrange equations is due to the robust function. In the inner fixed point iteration loop the Ψ' expressions are kept constant and are recomputed after each iteration. This finally leads to the following linear equations:

$$\begin{aligned} 0 = &\Psi'((I_z^{l,k+1})^2)(I_z^{l,k} + I_x^{l,k}\delta u^k + I_y^{l,k}\delta v^k)I_x^{l,k} \\ &+c\Psi'((I_z^{r,k+1})^2)(I_z^{r,k} + I_x^{r,k}(\delta u^k + \delta d'^k) + I_y^{r,k}\delta v^k)I_x^{r,k} \\ &-\lambda \operatorname{div}\left(\Psi'(\lambda|\nabla u^{k+1}|^2 + \lambda|\nabla v^{k+1}|^2 + \gamma|\nabla d'^{k+1}|^2)\nabla(u^k + \delta u^k)\right) \end{aligned} \tag{13}$$

$$\begin{aligned} 0 = &\Psi'((I_z^{l,k+1})^2)(I_z^{l,k} + I_x^{l,k}\delta u^k + I_y^{l,k}\delta v^k)I_y^{l,k} \\ &+c\Psi'((I_z^{r,k+1})^2)(I_z^{r,k} + I_x^{r,k}(\delta u^k + \delta d'^k) + I_y^{r,k}\delta v^k)I_y^{r,k} \\ &-\lambda \operatorname{div}\left(\Psi'(\lambda|\nabla u^{k+1}|^2 + \lambda|\nabla v^{k+1}|^2 + \gamma|\nabla d'^{k+1}|^2)\nabla(v^k + \delta v^k)\right) \end{aligned} \tag{14}$$

$$\begin{aligned} 0 = &c\Psi'((I_z^{r,k+1})^2)(I_z^{r,k} + I_x^{r,k}(\delta u^k + \delta d'^k) + I_y^{r,k}\delta v^k)I_x^{r,k} \\ &+c\Psi'((I_z^{d,k+1})^2)(I_z^{d,k} + I_x^{r,k}\delta d'^k)I_x^{r,k} \\ &-\gamma \operatorname{div}\left(\Psi'(\lambda|\nabla u^{k+1}|^2 + \lambda|\nabla v^{k+1}|^2 + \gamma|\nabla d'^{k+1}|^2)\nabla(d'^k + \delta d'^k)\right) \end{aligned} \tag{15}$$

where we omitted the iteration index of the inner fixed point iteration loop to keep the notation uncluttered. Expressions with iteration index $k+1$ are computed using the current increments $\delta u^k, \delta v^k, \delta d'^k$. We see that some terms of the original Euler-Lagrange equations for the have vanished as we have made use of $I(x_d, y, t)^r = I(x, y, t)^l$ in the linearised third constraint (Equation 4). After discretisation, the corresponding linear system is solved via successive over-relaxation. It is worth noting that, for efficiency reasons, it is advantageous to update the Ψ' after a few iterations of SOR. The shares of computation time taken by the different operations are shown in Figure 2.

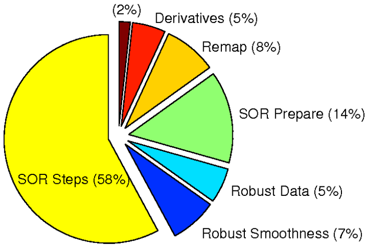

Image Size (pixels)	SOR Steps per Warp	Warps per Level	Total Time
640 × 480	45	2	975 ms
512 × 512	20	3	756 ms
320 × 240	45	2	205 ms

Fig. 2. Break down of computational time for our algorithm (3.0GHz Intel®Core$^{\text{TM}}$2). The pie graph shows the time distribution for the 640 × 480 images. The real-time applicability of the algorithm for image sizes of (320 × 240) is indicated in the table.

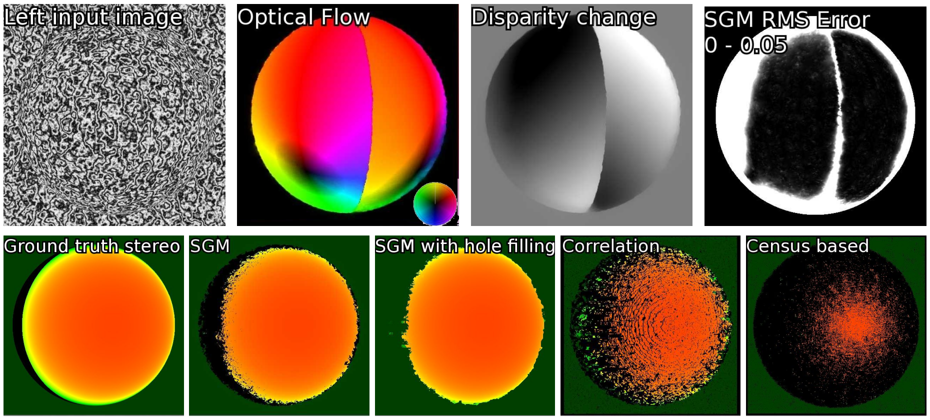

Fig. 3. Ground truth test: *rotating sphere*. Quantitative results are shown in Table 1. **Top:** optical flow and disparity change are computed on the basis of SGM stereo [5]. Colour encodes the direction of the optical flow (key in bottom right), intensity its magnitude. Disparity change is encoded from black (increasing) to white (decreasing). Bright parts of the RMS figure indicate high $RMS_{u,v,d'}$ error values of the computed scene flow. **Bottom:** disparity images are colour encoded green to orange (low to high). Black areas indicate missing disparity estimates or occluded areas.

5 Results

To assess the quality of our scene flow algorithm, it was tested on synthetic sequences, where the ground truth is known[5]. In a second set of experiments, we used real images to demonstrate the accuracy and practicality of our algorithms under real world conditions.

[5] The authors would like to thank Huguet *et al.* for providing their *sphere* scene.

Synthetic scenes. The first ground truth experiment is the *rotating sphere* sequence from [7] depicted in Figure 3. In this sequence the spotty sphere rotates around its y-axis to the left, while the two hemispheres of the sphere rotate in opposing vertical directions. The resolution is 512×512 pixels.
We tested the scene flow method together with four different stereo algorithms: semi-global matching (SGM [5]), SGM with hole filling (favours smaller disparities), correlation pyramid stereo [3], and an integer accurate census-based stereo algorithm [15]. The ground truth disparity was also used for comparison. For each stereo algorithm, we calculated the absolute angular error (AAE) and the root mean square (RMS) error

$$RMS_{u,v,d,d'} := \sqrt{\frac{1}{n}\sum_{\Omega}\|(u_i, v_i, d_i, d'_i)^\top - (u^*_i, v^*_i, d^*_i, d'^*_i)^\top\|^2} \tag{16}$$

where a superscript * denotes the ground truth solution and n is the number of pixels. In our notation for RMS, if a subscript is omitted, then both the respective ground truth and estimated value are set to zero. The errors were calculated in using two types of Ω: firstly, calculating statistics over all non-occluded areas, and secondly calculating over the whole sphere. As in [7], pixels from the background were not included in the statistics.

The smoothing parameters were set to $\lambda = 0.2$ and $\gamma = 2$. We used 60 SOR iterations at each pyramid level, resulting in an average runtime of 756

Stereo	RMS_d	Without occluded areas			With occluded areas		
Algorithm	(density)	$RMS_{u,v}$	$RMS_{u,v,d'}$	$AAE_{u,v}$	$RMS_{u,v}$	$RMS_{u,v,d'}$	$AAE_{u,v}$
Huguet *et al.* [7]	3.8 (100%)	0.37	0.83	1.24	0.69	2.51	1.75
Flow Only	$d' = 0$ for evaluation	0.34	1.46	1.26	0.67	2.85	1.72
Flow Only*		0.30	1.46	0.95	0.64	2.85	1.36
Ground truth		0.33	0.58	1.25	0.67	2.40	1.78
Ground truth*		0.31	0.56	0.91	0.65	2.40	1.40
SGM [5]	2.9 (87%)	0.35	0.64	1.33	0.66	2.45	1.82
SGM*		0.34	0.63	1.04	0.66	2.45	1.50
Fill-SGM	10.9 (100%)	0.43	0.75	2.18	0.77	2.55	2.99
Fill-SGM*		0.45	0.76	1.99	0.77	2.55	2.76
Correlation [3]	2.6 (43%)	0.34	0.75	1.31	0.67	2.51	1.84
Correlation*		0.33	0.73	1.02	0.65	2.50	1.52
Census based [15]	7.8 (16%)	0.36	1.08	1.30	0.67	2.65	1.75
Census based*		0.32	1.14	1.01	0.65	2.68	1.43

Table 1. Root mean square (pixels) and average angular error (degrees) for scene flow of the *rotating sphere* sequence. Various stereo algorithms are used as input for our scene flow estimation, generating varying results. A * denotes running until convergence. SGM (highlighted) is the best solution for its speed to accuracy ratio. "Flow Only" does not include stereo correspondences, thus calculates 2D optical flow only. For the evaluation we used the formula $AAE_{u,v} := \frac{1}{n}\sum_{\Omega}\arctan\left(\frac{uv^*-u^*v}{uu^*+vv^*}\right)$ to calculate the error to the ground truth flow (u^*, v^*) as used in [7].

milliseconds. Additionally, we let the algorithm run to convergence (change in $RMS \leq \varepsilon$) for better accuracy without changing λ and γ; this increased the computational time to around 3 seconds.

The resulting summary can be seen in Table 1. We achieve lower errors than the Huguet *et al.* method, when we let the method converge. Particularly, the RMS error of the scene flow is much smaller and we are still considerably faster. This is explained by the higher flexibility in choosing the disparity estimation method. Furthermore, we achieve real-time performance with little loss in accuracy. The table shows that SGM with hole filling yields inferior results than the other stereo methods. This is due to false disparity measurements in the occluded area. It is better to feed the sparse measurements of SGM to the variational framework, which yields dense estimates as well, but with higher accuracy. SGM was used in the remainder of the results section, as it gives the best results and is available on dedicated hardware without any extra computational cost.

In a second ground truth example we use a Povray-rendered traffic scene. The scene includes common features such as mixture of high and low textured areas on the ground plane and background, occlusions, reflections, and transparency of car windscreens. The vehicle in front of the camera (and its shadow) moves straight ahead. There are vehicles entering the main road from the left and the right. The camera system is also moving orthogonal to the image plane. We calculated the $RMS_{u,v,d'}$ error and the 4D angular error defined by:

$$AAE_{4D} := \frac{1}{n} \sum_{\Omega} \arccos \left(\frac{uu^* + vv^* + d'd'^* + 1}{\sqrt{(u^2 + v^2 + d'^2 + 1)\left((u^*)^2 + (v^*)^2 + (d'^*)^2 + 1\right)}} \right)$$

Results are shown in Figures 4 and 5. They compare favourably to the results obtained when running the code from [7]. The average $RMS_{u,v,d'}$ error for the whole sequence (subregion as in Figure 4) was 0.64 px and the 4D angular error was 3.01°. The sequence will be made publicly available in the internet to allow evaluation of future scene flow algorithms.

Real world scenes. Figure 6 and Figure 7 show scene flow results in real world scenes with a moving camera. A result video of the scene shown in Figure 1 is included in the supplemental material. Ego motion of the camera is known from vehicle odometry and compensated in the depicted results.

Figure 6 shows an image from a sequence where the ego-vehicle is driving past a bicyclist. The depicted scene flow shows that most parts of the scene, including the vehicle stopping at the traffic lights, are correctly estimated as stationary. Only the bicyclist is moving and its motion is accurately estimated.

Figure 7 shows results from a scene where a person runs from behind a parked vehicle. The ego-vehicle is driving forward at 30 km/h and turning to the left. The measurements on the ground plane and in the background are not shown to focus visual attention on the person. The results show that points on the parked vehicle are estimated as stationary, where as points on the person are registered as moving. The accurate motion results can be well observed for the person's legs, where the different velocities of each leg are well estimated.

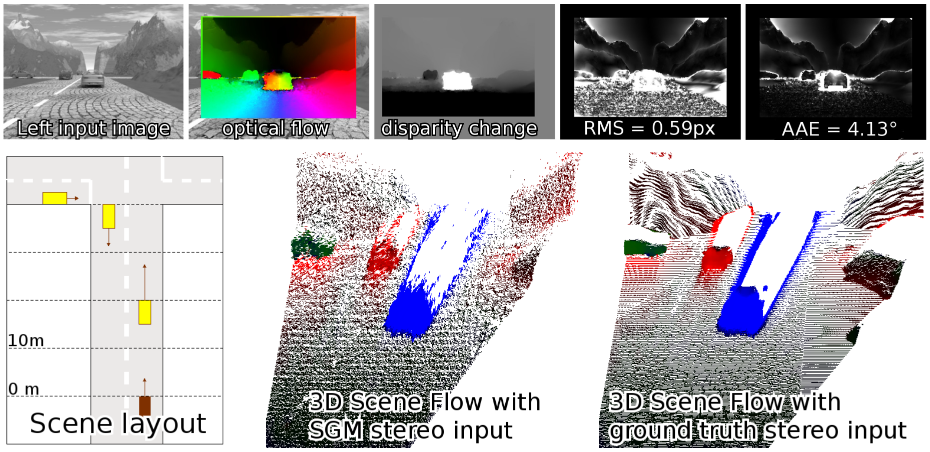

Fig. 4. Povray-rendered traffic scene (Frame 11). **Top:** Colour encodes direction (border = direction key) and intensity the magnitude of the optical flow vectors. Brighter areas in the error images denote larger errors. For comparison, running the code from [7] generates an RMS error of 0.91px and AAE of 6.83°. **Bottom right:** 3D views of the scene flow vectors. Colour encodes their direction and brightness their magnitude (black = stationary). The results from the scene are clipped at a distance of 100m. Accurate results are obtained even at greater distances.

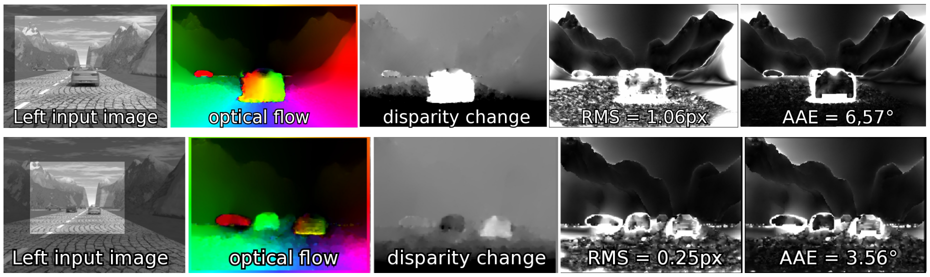

Fig. 5. More frames from the traffic scene in Figure 4. The top row highlights the problems such as transparency of the windshield, reflectance, and moving shadows. The bottom row demonstrates that we still maintain accuracy at distances of 50 m.

6 Conclusions

We presented a variational framework for dense scene flow estimation, which is based on a decoupling of the disparity estimation from the velocity estimation, while enforcing consistent motion in all involved images. We showed that this strategy has two main advantages: Firstly, we can choose optimal methods for estimating both disparity and velocity. In particular, we can combine occlusion handling and global (combinatorial) optimisation for disparity estimation with dense, sub-pixel accurate velocity estimation. Secondly, for the first

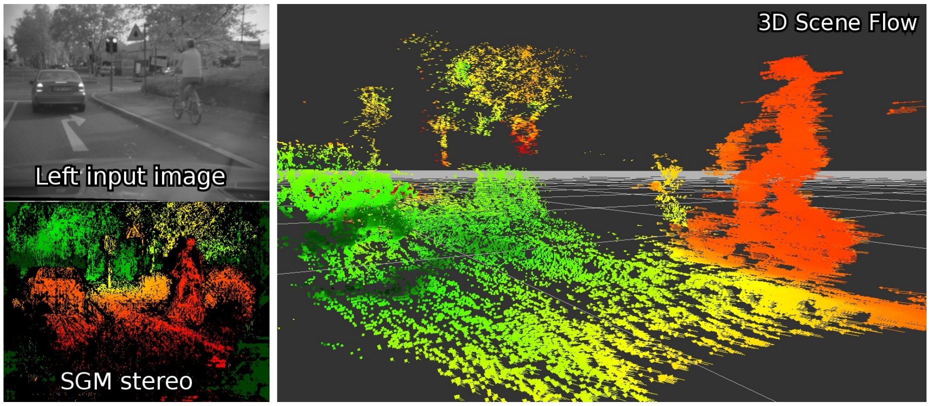

Fig. 6. Dense scene flow in a traffic scene. The colour in the lower left image encodes distance from red to green (close to far); the colour in the scene flow image (right) shows vector lengths after ego-motion compensation (green to red = 0 to $0.4m/s$). Only the cyclist is moving.

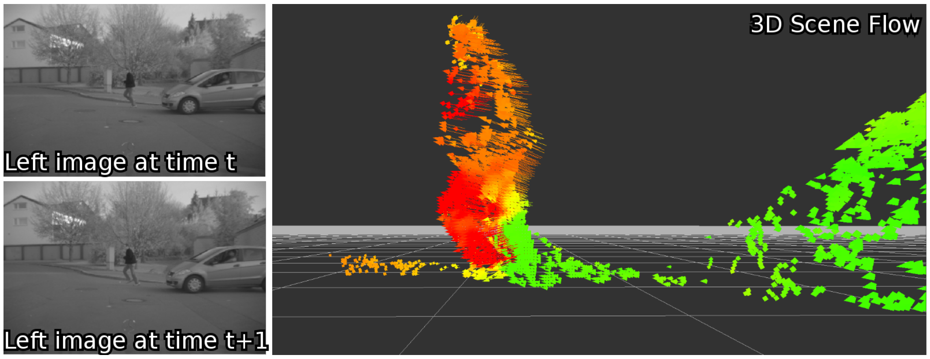

Fig. 7. Scene with a person running from behind a parked vehicle. The colour encoding is as in Figure 6.

time, we obtain dense scene flow results very efficiently in real-time. We showed that the approach works well on both synthetic and real sequences, and that it provides highly accurate velocity estimates, which compare favourably to the literature. Ongoing work will include temporal consistency by employing, for instance, Kalman filters. Another interesting aspect is the segmentation of the 3D velocity field.

References

1. T. Brox, A. Bruhn, N. Papenberg, and J. Weickert. High accuracy optical flow estimation based on a theory for warping. In *European Conference on Computer Vision (ECCV)*, pages 25–36, 2004.

2. A. Bruhn, J. Weickert, T. Kohlberger, and C. Schnörr. Discontinuity preserving computation of variational optic flow in real-time. In *ScaleSpace05*, pages 279–290, 2005.
3. U. Franke and A. Joos. Real-time stereo vision for urban traffic scene understanding. In *Proc. IEEE Intelligent Vehicles Symposium*, pages 273–278, Dearborn, 2000.
4. A. Fusiello, E. Trucco, and A. Verri. A compact algorithm for rectification of stereo pairs. *Machine Vision and Applications*, 12(1):16–22, 2000.
5. H. Hirschmüller. Stereo vision in structured environments by consistent semi-global matching. In *CVPR (2)*, pages 2386–2393. IEEE Computer Society, 2006.
6. B. Horn and B. Schunck. Determining optical flow. *Artificial Intelligence*, 17:185–203, 1981.
7. F. Huguet and F. Devernay. A variational method for scene flow estimation from stereo sequences. In *IEEE Eleventh International Conference on Computer Vision, ICCV 07, Rio de Janeiro, Brazil*, October 2007.
8. M. Isard and J. MacCormick. Dense motion and disparity estimation via loopy belief propagation. In *ACCV (2)*, volume 3852 of *Lecture Notes in Computer Science*, pages 32–41. Springer, 2006.
9. E. Mémin and P. Pérez. Dense estimation and object-based segmentation of the optical flow with robust techniques. *IEEE Transactions on Image Processing*, 7(5):703–719, May 1998.
10. D. Min and K. Sohn. Edge-preserving simultaneous joint motion-disparity estimation. In *ICPR '06: Proc. 18th International Conference on Pattern Recognition*, pages 74–77, Washington, DC, USA, 2006. IEEE Computer Society.
11. I. Patras, E. Hendriks, and G. Tziritas. A joint motion/disparity estimation method for the construction of stereo interpolated images in stereoscopic image sequences. In *Proc. 3rd Annual Conference of the Advanced School for Computing and Imaging*, Heijen, The Netherlands, 1997.
12. M. Pollefeys, R. Koch, and L. V. Gool. A simple and efficient rectification method for general motion. In *IEEE International Conference on Computer Vision*, pages 496–501, 1999.
13. J.-P. Pons, R. Keriven, and O. Faugeras. Multi-view stereo reconstruction and scene flow estimation with a global image-based matching score. *Int. J. Comput. Vision*, 72(2):179–193, 2007.
14. C. Rabe, U. Franke, and S. Gehrig. Fast detection of moving objects in complex scenarios. In *Proc. IEEE Intelligent Vehicles Symposium*, pages 398–403, June 2007.
15. F. Stein. Efficient computation of optical flow using the census transform. In *Proc. DAGM (Pattern Recognition)*, pages 79–86, 2004.
16. C. Tomasi and T. Kanade. Detection and tracking of point features. Technical Report CMU-CS-91-132, Carnegie Mellon University, April 1991.
17. S. Vedula, S. Baker, P. Rander, R. Collins, and T. Kanade. Three-dimensional scene flow. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(3):475 – 480, March 2005.
18. C. Zach, T. Pock, and H. Bischof. A duality based approach for realtime tv-L^1 optical flow. In *Proc. DAGM (Pattern Recognition)*, pages 214–223, 2007.
19. Y. Zhang and C. Kambhamettu. On 3d scene flow and structure estimation. In *Proc. IEEE Conf. in Computer Vision and Pattern Recognition*, volume 2, page 778, Los Alamitos, CA, USA, 2001. IEEE Computer Society.