Dense, Robust, and Accurate Motion Field Estimation from Stereo Image Sequences in Real-time

Clemens Rabe, Thomas Müller, Andreas Wedel, and Uwe Franke

Daimler Research, Sindelfingen

Abstract. In this paper a novel approach for estimating the three dimensional motion field of the visible world from stereo image sequences is proposed. This approach combines dense variational optical flow estimation, including spatial regularization, with Kalman filtering for temporal smoothness and robustness. The result is a dense, robust, and accurate reconstruction of the three-dimensional motion field of the current scene that is computed in real-time. Parallel implementation on a GPU and an FPGA yields a vision-system which is directly applicable in real-world scenarios, like automotive driver assistance systems or in the field of surveillance. Within this paper we systematically show that the proposed algorithm is physically motivated and that it outperforms existing approaches with respect to computation time and accuracy.

1 Introduction

Estimating the three-dimensional motion vector field from stereo image sequences remains as one of the fundamental computational challenges and is a key task in computer vision. Different variants of this problem arise in the estimation of ego motion [1], object motion [2], human motion [3], and motion segmentation [4]. Knowledge of the surrounding motion field is the basis for a wide range of technical applications, e.g. in automotive driver assistance systems or in the field of surveillance. Therefore, and especially in safety relevant applications, robustness, density of information, accuracy as well as real-time capability are of utmost importance.

Some approaches known from literature use sparse feature-based tracking methods and increase the robustness by temporal integration using filters. Others apply space-related regularizers for a dense computation from only two consecutive frames. In this paper, we combine such a dense variational approach on the image domain with Kalman filters at every single pixel to establish temporal smoothness of the dense three-dimensional motion field.

In this paper, we present a new algorithm called *Dense6D*, that estimates the motion field by fusing dense stereo and optical flow information. A real world example of such an estimate is shown in Fig. 1. An improvement in accuracy and robustness with respect to standard approaches known from literature is also achieved with *Variational6D*, which replaces the optical flow component

Fig. 1. *left:* typical traffic scene. *right:* motion field, estimated by the Dense6D algorithm proposed in this paper. The color encodes the velocity (from green to red) of the observed points.

of Dense6D by a variational scene flow method. Throughout the paper, the term *scene flow* will denote the three-dimensional motion field consisting of the optical flow and the disparity change along the optical flow vectors between two consecutive frames.

1.1 Related Work

Due to the importance of the problem, a lot of different approaches to image based motion field estimation have been proposed in the last three decades. Most of them can be classified into the following main strategies:

- model based approaches
- sparse feature tracking methods using multiple image frames
- dense scene flow computation from two consecutive frames

The estimation of motion vectors involves the reconstruction of the three dimensional scene via stereo matching and the estimation of point correspondences between two or more consecutive images. Both problems are classical ill-posed problems in the sense that merely matching of similar intensities will typically not give rise to a unique solution. The three mentioned strategies choose different ways to overcome the ill-posedness.

The model based approaches, like in [3] or [2] use physically constrained object or human models to make the problem well-posed; however, the need of a model disqualifies the model based approaches in a large variety of situations.

The feature tracking and scene flow appraoches use regularization to make the problem well-posed. This regularization is either formulated in the time domain for the tracking of features in [5] or [6] or in the spatial domain, imposing smoothness of the motion field between two consecutive frames like in [7] or [8]. In this paper we revisit both, feature tracking and dense scene flow computation,

and suggest the use of Kalman filters for every image pixel to reconstruct a dense and robust three-dimensional motion field of the depicted scene.

Scene flow computation methods are mainly based on the classic optical flow algorithm by Horn and Schunck in [9], where the flow field is computed as the minimizer of an energy functional that assumes constant image intensities and a smooth flow field. This framework has been improved in [10] to cope with flow discontinuities and outliers and in [11] to cope with large flow vectors. In recent years, several real-time optical flow methods have been proposed, e.g. in [12],[13].

Joint motion and disparity estimation for the scene flow computation was introduced in [14]. In [8] the motion and disparity estimation steps were decoupled in order to achieve real-time capability without loosing accuracy.

On the other hand, the application of Kalman filters [15] in real-time motion field estimation was proposed in [16] and later as *6D-Vision* in [17], using the well-known KLT-tracker [5] and a dense stereo disparity field as input. However, this method only yields sparse information. To build a vision system which provides a dense, robust and accurate motion field in real-time, we suggest to replace the tracker by a dense variational optical or scene flow algorithm. Despite the computational complexity and the large volume of data, special computation schemes for the filtering process, implemented on modern graphic processing units (GPUs), are used to ensure real-time capability (25 Hz in our implementation).

1.2 Outline

In Chapter 2 we will shortly revisit some approaches to optical flow and scene flow computation, which are then used as input for the filtering process. In Chapter 3 we will introduce the concept of Kalman filters for motion estimation and the novel dense filtered motion estimation approaches based on stereo, optical flow and scene flow. Chapter 4 will experimentally demonstrate the new approaches and systematically evaluate the gain in robustness and accuracy over existing methods. Chapter 5 concludes this contribution with a summary and an outlook on future work.

2 Two-Frame Motion Field Estimation

2.1 Combination of Optical Flow and Stereo

For the estimation of the motion field of the environment, we consider the two stereo image pairs $I_{\{1,2\}}^{\{\mathrm{L,R}\}} : \Omega \rightarrow \mathbb{R}$ on the image domain $\Omega = \{\boldsymbol{x}\} \subset \mathbb{R}^2$. Throughout this paper, we will assume that the camera system is calibrated, and the taken images are preprocessed by a rectification module that performs a lens-correction and establishes a standard stereo configuration.

Having determined the optical flow $\boldsymbol{u} : \Omega \rightarrow \mathbb{R}^2$, an inverse transformation together with known depth at both time instants can yield the three-dimensional motion field information of actual interest.

When searching the whole consecutives image for corresponding single gray values, so that (with $\boldsymbol{u} \equiv \boldsymbol{u}(\boldsymbol{x})$)

$$\rho(\boldsymbol{x}, \boldsymbol{u}) = I_2(\boldsymbol{x} + \boldsymbol{u}) - I_1(\boldsymbol{x}) = 0\,, \tag{1}$$

this obviously leads to an ill-posed problem. In [9] Horn and Schunck proposed to overcome this by minimizing a global energy functional on the whole image domain, consisting of a data consistency term and a regularization term,

$$E[\boldsymbol{u}] = \int_{\Omega} \left\{ \lambda \left| \rho(\boldsymbol{x}, \boldsymbol{u}(\boldsymbol{x})) \right|^n + \sum_{i=x,y} \left| \boldsymbol{\nabla} u_i(\boldsymbol{x}) \right|^n \right\} \mathrm{d}\boldsymbol{x} \tag{2}$$

with $n = 2$. The parameter $\lambda \in \mathbb{R}^+$ weights between the data and the regularization. This method leads to dense, accurate results and yields real-time performance on modern hardware.

The case $n = 2$ is quite simple to compute, but suffers from blurring effects around flow edges and over-weights outliers. Therefore, we use the computation method proposed by Wedel et al. in [18] based on [13], where $n = 1$ leads to improved results. This method solves the two terms in eq. (2) by introducing an additional coupling term and an iterative solution scheme on a coarse-to-fine grid. The data term part is solved directly point-wise by a thresholding step, while the regularizing smoothness term is solved by a dual approach, proposed by Chambolle in [19].

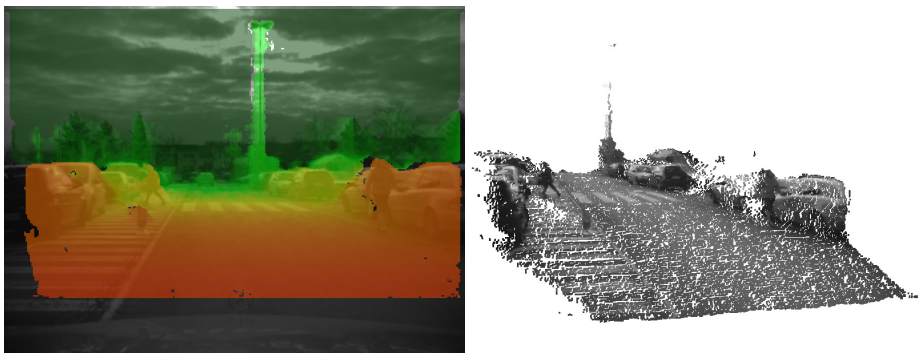


Fig. 2. *left:* traffic scene with SGM stereo computation. The color encodes the distance from near (red) to far (green). *right:* three-dimensional visualization of the corresponding scene.

In our work, the dense semi-global-matching (SGM) method by Hirschmüller [20] is used for the estimation of the disparity images $d_{\{1,2\}} : \Omega \rightarrow \mathbb{R}^+$ (see Fig. 2). The algorithm is available on dedicated parallel hardware (FPGA). Therefore, the disparity computation does not effect the real-time performance of our motion field estimation in a negative way.

With the knowledge of the optical flow field $\boldsymbol{u}(\boldsymbol{x})$ between two consecutive frames and both depth images $Z_{\{1,2\}}(\boldsymbol{x}) \sim 1/d_{\{1,2\}}(\boldsymbol{x})$, the three-dimensional motion field can be determined. However, this straight-forward differential approach usually leads to insufficient results, due to noisy depth measurements.

2.2 Variational Scene Flow

Noisy depth measurements lead to a noisy motion field estimation along the optical rays. To reduce this noise, we use a global variational approach where the disparity change is regularized together with the optical flow [8]. A decoupled depth measurement is used for the first frame and yields the disparity field $d_1(\boldsymbol{x})$, related to the left image, while the disparity change $\dot{d} : \Omega \to \mathbb{R}$ is estimated via a global optimization scheme together with the optical flow $\boldsymbol{u}$. The functional to minimize is defined as

$$E\left[\boldsymbol{u}, \dot{d}\right] = \int_{\Omega} \left\{ R\left(\boldsymbol{u}(\boldsymbol{x}), \dot{d}(\boldsymbol{x})\right) + \sum_{i=x,y} \left|\boldsymbol{\nabla} u_i(\boldsymbol{x})\right| + \left|\boldsymbol{\nabla} \dot{d}(\boldsymbol{x})\right| \right\} \mathrm{d}\boldsymbol{x} \tag{3}$$

with the data term

$$R\left(\boldsymbol{u}, \dot{d}\right) = \lambda_{\mathrm{L}} \left|\rho_{\mathrm{L}}(\boldsymbol{u})\right| + \lambda_{\mathrm{R}} \left|\rho_{\mathrm{R}}\left(\boldsymbol{u}, \dot{d}\right)\right| + \lambda_2 \left|\rho_2\left(\boldsymbol{u}, \dot{d}\right)\right| \tag{4}$$

and the residuals (with $d_1 \equiv d_1(\boldsymbol{x})$ and the unity vector in x direction $\boldsymbol{e}_x$)

$$\rho_{\mathrm{L}}(\boldsymbol{u}) = I_2^{\mathrm{L}}(\boldsymbol{x} + \boldsymbol{u}) - I_1^{\mathrm{L}}(\boldsymbol{x}) \tag{5}$$

$$\rho_{\mathrm{R}}\left(\boldsymbol{u}, \dot{d}\right) = I_2^{\mathrm{R}}\left(\boldsymbol{x} + \boldsymbol{u} - \left(d_1 + \dot{d}\right)\boldsymbol{e}_x\right) - I_1^{\mathrm{R}}(\boldsymbol{x} - d_1 \boldsymbol{e}_x) \tag{6}$$

$$\rho_2\left(\boldsymbol{u}, \dot{d}\right) = I_2^{\mathrm{R}}\left(\boldsymbol{x} + \boldsymbol{u} - \left(d_1 + \dot{d}\right)\boldsymbol{e}_x\right) - I_2^{\mathrm{L}}(\boldsymbol{x} + \boldsymbol{u})\ . \tag{7}$$

For the numerical computation, the data term and the regularizations are coupled by an additional term to establish an iterative solution scheme with a coarse-to-fine approach. The data term can then be solved point-wise by implementing $|x| \approx \sqrt{x^2 + \varepsilon}, \varepsilon \ll 1$, linearizing the residuals in Eqs. (5) - (7) and performing gradient descend steps. The regularization terms in the optical flow $\boldsymbol{u}$ as well as in the disparity change $\dot{d}$ are solved by the former mentioned dual approach. This method provides better results, compared to the approach with optical flow and stereo, but demands higher computational costs.

3 Temporal Integration of the Motion Field

To increase the robustness and accuracy of the estimated motion field, we suggest a temporal integration using Kalman filters. In this section, the underlying Kalman filter model is explained in detail for the proposed Dense6D and Variational6D algorithms.

3.1 Model

In our pinhole stereo camera system, the 3d structure of the observed scene is immediately reconstructed by a stereo algorithm. In this configuration, the left image point $\boldsymbol{x} = (x, y)^\top$ is the projection of a world point $\tilde{\boldsymbol{X}} = (X, Y, Z, 1)^\top$. Expressed in homogeneous coordinates, this relation is given by

$$\begin{pmatrix} x & y & d & 1 \end{pmatrix}^\top \backsimeq \boldsymbol{\Pi} \cdot \tilde{\boldsymbol{X}} \tag{8}$$

with the positive disparity $d \equiv d(\boldsymbol{x})$. The extended projection matrix $\boldsymbol{\Pi}$ is given as

$$\boldsymbol{\Pi} = \begin{pmatrix} f_x & 0 & x_0 & 0 \\ 0 & f_y & y_0 & 0 \\ 0 & 0 & 0 & b \cdot f_x \\ 0 & 0 & 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} \mathbf{R}_\mathrm{c} & \boldsymbol{t}_\mathrm{c} \\ \mathbf{0}^\top & 1 \end{pmatrix} \tag{9}$$

with f_x and f_y as the focal lengths in pixel, $(x_0, y_0)^\top$ as the principal point in pixel and b as the base width of the stereo camera system. The rotation matrix $\mathbf{R}_\mathrm{c}$ and the translation vector $\boldsymbol{t}_\mathrm{c}$ describe the extrinsic orientation of the camera system to the world coordinate system. To determine the three-dimensional world position for an observed image point $\boldsymbol{x}$ with known disparity d, eq. (8) has to be inverted.

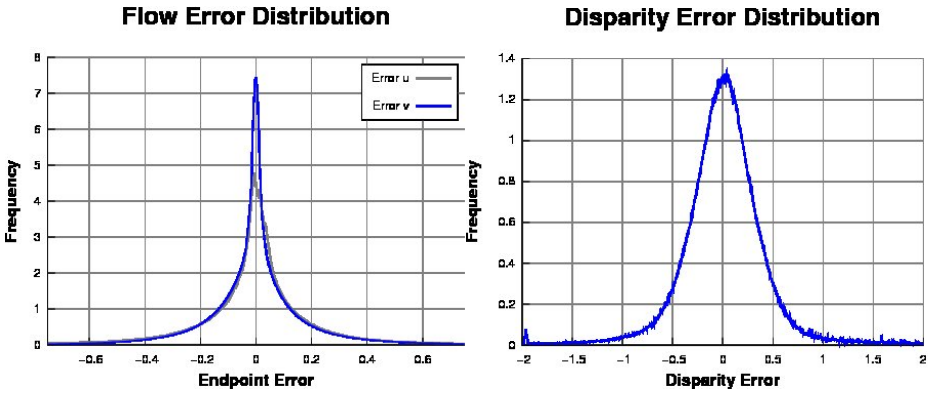


Fig. 3. *left:* Dense optical flow error distribution, *right:* SGM stereo error distribution, both related to ground truth data from synthetic sequences.

Having established a correspondence over time for an observed image point by an optical flow or feature tracking algorithm, the 3d motion can be calculated directly from the reconstructed 3d points. However, such an approach suffers heavily from the immanent measurement noise and thus does not yield robust results. Therefore, we present here a method to estimate the 3d position and 3d motion of a point using a Kalman filter [15]. Due to the recursive nature of

the Kalman filter, the estimation is improved continuously with each measurement, by updating the state vector and its associated covariance matrix. This eliminates the need to save a history of measurements and is computationally highly efficient. Additionally, Kalman filters provide measurement uncertainties which can be regarded when the motion field is evaluated for further applications. Looking at the error distributions of the input data shown in Fig. 3, the application of the Kalman filter is justified.

The state vector of the Kalman filter is defined as $\boldsymbol{\xi} = \left(X, Y, Z, \dot{X}, \dot{Y}, \dot{Z}\right)^\top$, the combination of the 3d position and the 3d velocity vector. The system model describes the propagation of the state vector $\boldsymbol{\xi}_t$ of the previous time step $t-1$ to the current one t and assumes a linear motion. It is given by the linear equation system

$$\tilde{\boldsymbol{\xi}}_t = \begin{pmatrix} \mathbf{R}_\mathrm{e} & \Delta t \cdot \mathbf{R}_\mathrm{e} \\ \mathbf{0} & \mathbf{R}_\mathrm{e} \end{pmatrix} \boldsymbol{\xi}_{t-1} + \begin{pmatrix} \boldsymbol{t}_\mathrm{e} \\ \mathbf{0} \end{pmatrix} \tag{10}$$

with $\mathbf{R}_\mathrm{e}$ and $\boldsymbol{t}_\mathrm{e}$ as the rotation and the translation components of the inverse motion of the observer, and Δt as the time between both frames.

The measurement model of the Kalman filter describes the relation between the measurement vector $\boldsymbol{z} = (x, y, d)^\top$ and the state vector $\boldsymbol{\xi}$. Here, only the position components of the state vector are directly measured, and the relation between the measured projection $\boldsymbol{z}$ and the reconstructed 3d point is given by eq. (8). Since the measurement model must be formulated in the euclidean space rather than the projective space, the measurement model is non-linear:

$$\tilde{\boldsymbol{z}} = w \cdot \begin{pmatrix} x & y & d & 1 \end{pmatrix}^\top = \boldsymbol{\Pi} \cdot \begin{pmatrix} X & Y & Z & 1 \end{pmatrix}^\top \tag{11}$$

$$\boldsymbol{z} = \frac{1}{w} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \cdot \tilde{\boldsymbol{z}} \tag{12}$$

Therefore, extended Kalman filters have to be applied.

3.2 Filtered Tracks and Stereo: 6D-Vision

In [6], 2000 Kanade-Lucas-Tomasi (KLT) features are used to generate measurements which are temporally integrated by Kalman filters with the model equations mentioned previously. The current measurement vector $\boldsymbol{z}_t$ is determined by

$$\boldsymbol{z}_t = \begin{pmatrix} \boldsymbol{x}_t \\ d_t\left(\boldsymbol{x}_t\right) \end{pmatrix}, \quad \boldsymbol{x}_t = \boldsymbol{x}_{t-1} + \boldsymbol{u}\left(\boldsymbol{x}_{t-1}\right) \tag{13}$$

with $\boldsymbol{u}\left(\boldsymbol{x}_{t-1}\right)$ as the optical flow related to the previous position $\boldsymbol{x}_{t-1}$ of the feature, computed for example by the Lucas-Kanade method [5], and the corresponding disparity $d_t(\boldsymbol{x}_t)$ at the new image position $\boldsymbol{x}_t$.

Note that $\boldsymbol{x}_{t-1}$ in eq. (13) depicts the old *measured* image position at the previous frame, not the projection of the filtered state $\boldsymbol{\xi}_{t-1}$. That means the

image position of the features is only determined by the feature tracker, while the filtering only influences the velocity and the disparity estimation. This way, undesired low pass filtering effects of the Kalman filter are avoided. Together with a multiple-filter approach, that reduces the settling time of a filter by running multiple differently initialized filters in parallel, the 6D-Vision motion field estimation method provides robust results in real-time, even in real world scenarios.

3.3 Filtered Dense Optical Flow and Stereo: Dense6D

The information provided by the 6D-Vision approach is only sparse. However, to utilize as much information as possible from a stereo image sequence, we replace the feature tracker in the measurement step by a dense optical flow algorithm (*Dense6D*). Modern parallel hardware, an NVIDIA graphics adapter with CUDA capability in our implementation, together with sophisticated numerical computation schemes at the filtering process, enables us to assign Kalman filters to every single pixel of the input image sequence (of the size $640\,\text{px} \times 480\,\text{px}$) and to apply them in real-time (at $25\,\text{Hz}$). For numerical stability, the implementation is based on the well-known U-D factorization proposed by Bierman et al. [21]. A code generator takes advantage of the sparse measurement matrix and produces the actual CUDA implementation.

At the beginning of the computation step from image I_{t-1} to I_t, every pixel $\boldsymbol{x}_{t-1}$ on the discrete pixel grid is associated with one Kalman filter $\mathcal{K}_{t-1}(\boldsymbol{x}_{t-1})$ and one sub-pixel component $\boldsymbol{s}_{t-1}(\boldsymbol{x}_{t-1})$) After having determined the dense optical flow field from I_{t-1} to I_t, and after having updated the filters during the filtering step, $\mathcal{K}_{t-1}(\boldsymbol{x}_{t-1}) \rightarrow \mathcal{K}_t(\boldsymbol{x}_{t-1})$, the updated Kalman filter field $\mathcal{K}_t(\boldsymbol{x}_{t-1})$ must be warped along the sub-pixel accurate optical flow $u(\boldsymbol{x}_{t-1})$, to receive the filter field $\mathcal{K}_t(\boldsymbol{x}_t)$ on the new discrete pixel positions $\boldsymbol{x}_t$. The updates of the positions and the sub-pixel components are given by

$$\boldsymbol{x}_t = \lfloor \boldsymbol{x}_{t-1} + \boldsymbol{s}_{t-1}(\boldsymbol{x}_{t-1}) + \boldsymbol{u}(\boldsymbol{x}_{t-1}) + 0.5\ \text{px} \rfloor \tag{14}$$

$$\boldsymbol{s}_t(\boldsymbol{x}_t) = [\boldsymbol{s}_{t-1}(\boldsymbol{x}_{t-1}) + u(\boldsymbol{x}_{t-1}) + 0.5\ \text{px}] \bmod 1\ \text{px} - 0.5\ \text{px} \tag{15}$$

At every time step the sub-pixel component is updated due to the sub-pixel accurate optical flow, which is always taken from the discrete position of the pixel grid, since exact optical flow information is only available at these points.

During the resampling step it is possible that not every pixel $\boldsymbol{x}_t$ of the current image is referred by a flow vector $\boldsymbol{u}(\boldsymbol{x}_{t-1})$. In this case a new filter has to be created with predefined initial values and associated with the empty pixel. Another option is to initialize the filter based on the states and the covariances of the surrounding filters.

On the other hand, if one pixel $\boldsymbol{x}_t$ of the current image is referred by more than one flow vectors $\boldsymbol{u}(\boldsymbol{x}_{t-1})$, one either has to decide which one of the filters will be used with the corresponding pixel for the next frame, or has to combine them to a new one. In this case, the covariances of the concurring filters can weight between them. It is also reasonable to use the depth information, so that

the filter with the smallest Z value in the filter state can survive, while the other ones are deleted.

For performance reasons, our implementation does not perform such an extended analysis, but generates the target image on a first-come first-serve basis.

3.4 Filtered Variational Scene Flow

If the dense optical flow method is replaced by a variational scene flow scheme as proposed in sec. 2.2, the estimation of the disparity change $\dot{d}(\boldsymbol{x})$ from eq. (3) can be used as an additional measurement. In this case, the measurement vector for the Kalman filter is

$$\boldsymbol{z}_t = \begin{pmatrix} \boldsymbol{x}_t \\ d_t(\boldsymbol{x}_t) \\ d_{t-1}(\boldsymbol{x}_{t-1}) + \dot{d}(\boldsymbol{x}_{t-1}) \end{pmatrix}, \tag{16}$$

while the new matrix

$$\mathbf{H} = \begin{pmatrix} f_x & 0 & x_0 & 0 \\ 0 & f_y & y_0 & 0 \\ 0 & 0 & 0 & b \cdot f_x \\ 0 & 0 & 0 & b \cdot f_x \\ 0 & 0 & 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} \mathbf{R}_\mathrm{c} & \boldsymbol{t}_\mathrm{c} \\ \mathbf{0}^\top & 1 \end{pmatrix} \tag{17}$$

replaces the extended projection matrix $\mathbf{\Pi}$ in Eq. (11). The Kalman filter weights between the two disparity measurements regarding the measurement covariance matrix.

4 Evaluation

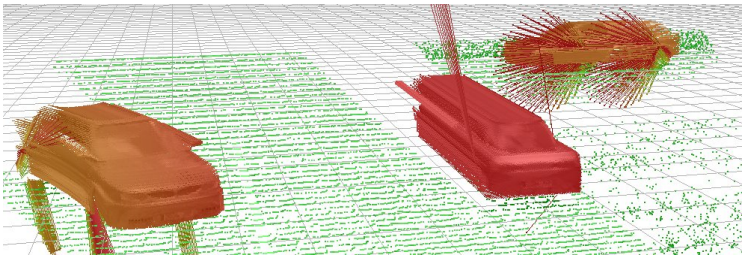
In our experiments, we compare the following motion field estimation techniques described in this paper:

1. Differential motion field estimation from optical flow and stereo (Sec. 2.1)
2. Variational scene flow from two frames (Sec. 2.2)
3. the Kalman filtered method, using dense optical flow and stereo (Dense6D, introduced in Sec. 3.3)
4. the filtered variational scene flow approach (Variational6D, introduced in Sec. 3.4).

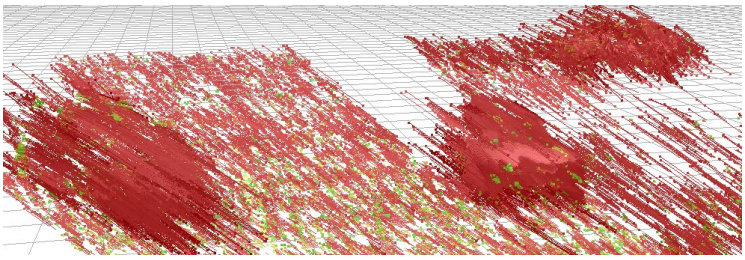
4.1 Evaluation with Ground Truth Information

In the first experimental part, we analyze our vision system on a synthetic stereo image sequence rendered with Povray [22]. The experiments are conducted on a sequence with an image resolution of 640 px × 480 px × 12 bit and 150 frames.

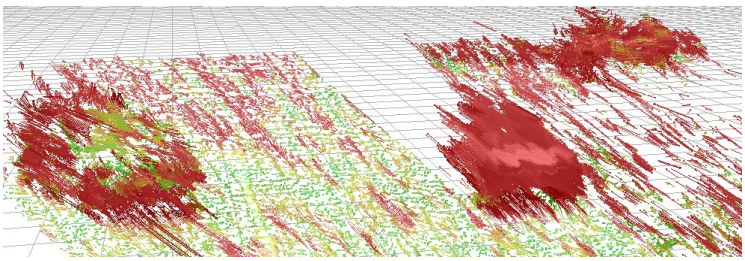
The evaluation platform consists of an Intel Quad-Core 3 GHz processor and an NVIDIA GeForce 285 GTX graphics adapter. On this configuration, the dense



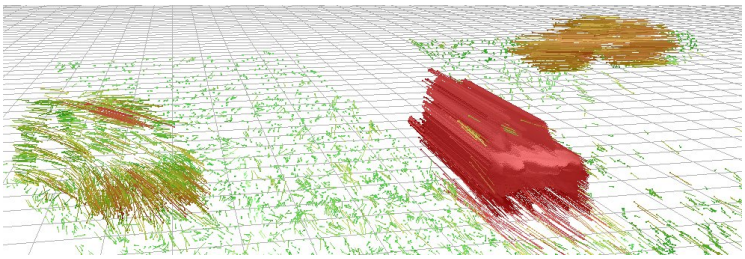
(a) Ground truth



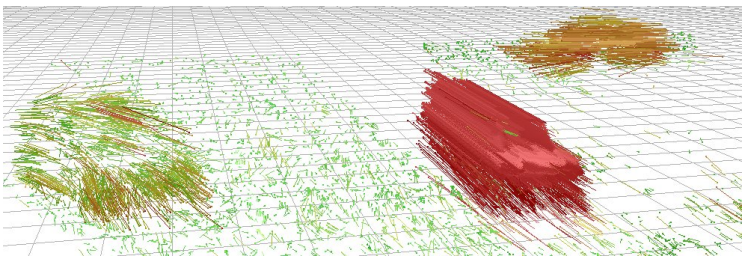
(b) Direct combination of optical flow and stereo



(c) Scene flow

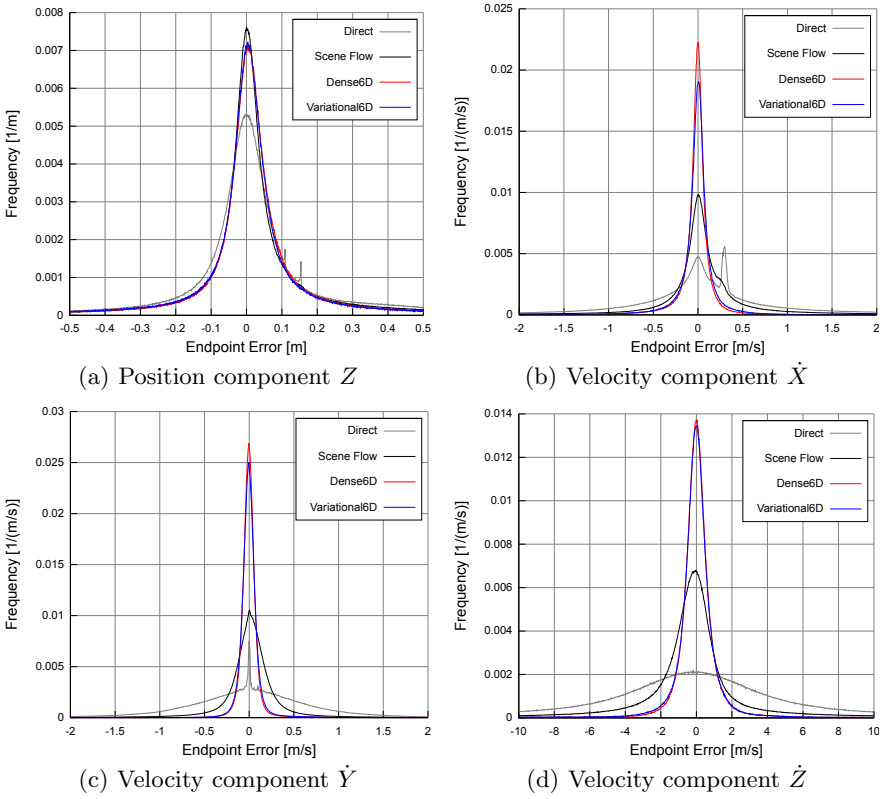


(d) Dense6D



(e) Variational6D

Fig. 4. Estimated motion field of the described methods. The color encodes the velocity: green encodes 0.0 m/s, red encodes 8.0 m/s. The vectors show the predicted 3d position in 0.250 s (Figures (a), (d), (e)) resp. 0.050 s (Figures (b), (c)).



(a) Position component Z (b) Velocity component $\dot{X}$

(c) Velocity component $\dot{Y}$ (d) Velocity component $\dot{Z}$

Fig. 5. Error distributions of the Z position and the velocity components calculated from the direct combination of optical flow and stereo (gray), the scene flow (black), the Dense6D method (red), and the Variational6D method (blue).

	Z [m]		$\dot{X}$ [m/s]		$\dot{Y}$ [m/s]		$\dot{Z}$ [m/s]	
Method	ME	RMS	ME	RMS	ME	RMS	ME	RMS
Direct	0.0010	2.749	0.0462	42.0093	0.0004	15.370	0.4374	141.442
Scene Flow	0.0080	2.807	0.0179	22.7186	0.0172	11.470	-0.1173	67.520
Dense6D	0.0104	1.068	-0.0065	0.3623	-0.0044	0.339	0.0107	2.538
Variational6D	0.0085	1.282	-0.0007	0.3712	-0.0040	0.319	-0.0044	2.537

Table 1. Median error (ME) and root mean square error (RMS) of the Z position and velocity components of the four evaluated methods.

optical flow calculation is performed in 24 ms, whereas the dense scene flow computation takes 65 ms. The 640×480 Kalman filters are processed in 12 ms. This enables us to achieve a framerate of 25 Hz for the Dense6D algorithm and about 10 Hz for the Variational6D approach.

In our exemplary sequence, the camera moves through an artificial traffic scene containing crossing and turning vehicles. Fig. 4 shows the motion vector fields for one frame of the sequence. The left vehicle performs a turning maneuver, while the remaining two cars are moving linearly. The vehicle in the middle moves at a constant speed, whereas the car coming from the right performs a constant deceleration. Obviously, the Dense6D and Variational6D algorithms outperform the differential approaches.

Since the three-dimensional ground truth position and motion fields are available, the error distributions for $Z, \dot{X}, \dot{Y}$ and $\dot{Z}$ can be determined. Accumulated over the whole image Ω and the whole sequence $[0, T]$, the error distributions are shown in Fig. 5. In addition, the median (ME) of the error distribution and the root mean squared (RMS) error is computed for the quantities $Z, \dot{X}, \dot{Y}$ and $\dot{Z}$.

One can clearly see from Fig. 5 and Tab. 1 that the proposed Dense6D algorithm outperformes the scene flow computation method with respect to accuracy and robustness.

Overall, we are not able to detect significant advantages regarding the accuracy of one of the new proposed algorithms against the other one in this paper. Therefore, due to the less complex computation scheme of the Dense6D algorithm against Variational6D, we suggest to use the first one in future motion field estimation tasks.

However, as one can see from the results presented in this chapter, both new approaches outperform dense real-time methods known from literature by far.

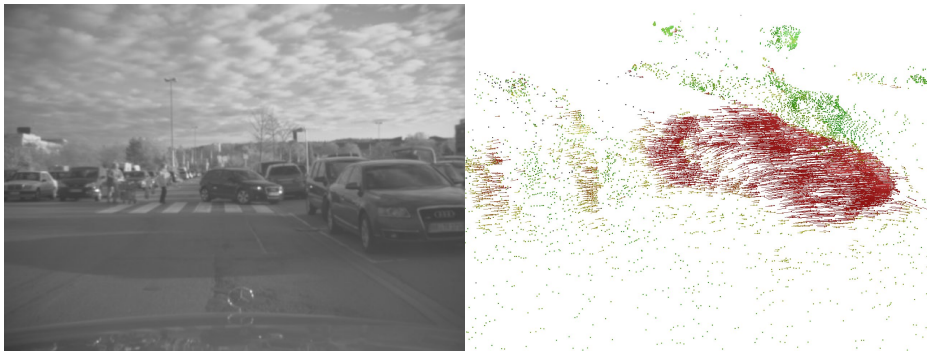


Fig. 6. *left:* typical traffic scene. *right:* corresponding 3d motion field, estimated by the Dense6D algorithm proposed in this paper. The color encodes the velocity (from green to red) of the observed points.

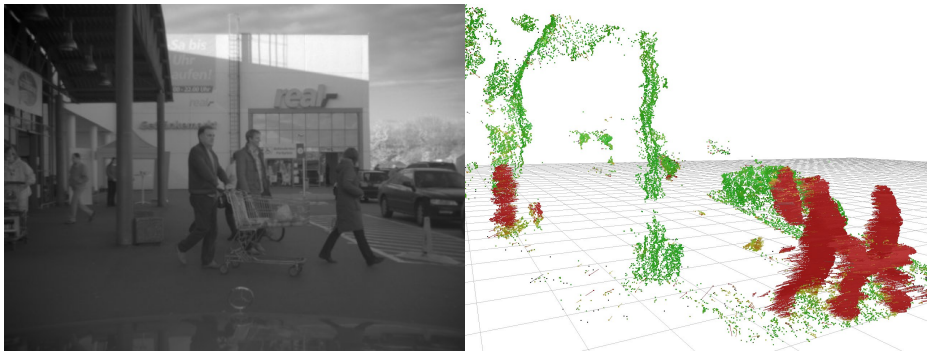


Fig. 7. *left:* typical traffic scene. *right:* corresponding 3d motion field, estimated by the Dense6D algorithm proposed in this paper. The color encodes the velocity (from green to red) of the observed points.

4.2 Real World Results

The two proposed new estimation methods are directly applicable in real-world scenarios, being able to build the basis for robust reliable object detection and segmentation. Fig. 6 shows the estimated motion vector field of a turning vehicle at a distance of about 30 m. Here, the observer was moving at a speed of about 3 $^{\mathrm{m}}/_{\mathrm{s}}$. In Fig. 7, the motion field of multiple pedestrians is shown. Here, the observer was also moving at about 1 $^{\mathrm{m}}/_{\mathrm{s}}$, including a strong turning maneuver. For visualization purposes, the motion of the camera was compensated using inertial sensor data. Dense6D is currently implemented in our research car and can become a key element in future safety driver assistance systems.

5 Conclusions

In this paper, we have proposed two approaches to dense, robust, and accurate motion field estimation in real-time. We have combined dense variational optical or scene flow estimation techniques with Kalman filters, assuming a linear motion model. Evaluation of the relevant error quantities compared to synthetic ground truth data show that these approaches lead to far better results in real-time than known by literature so far. This is apparently similar in real world scenarios.

The next future work will include a multi-filter implementation on the GPU and the consideration of flow uncertainties in the filtering process.

References

1. Badino, H.: A robust approach for ego-motion estimation using a mobile stereo platform. In: IWCM'04, Günzburg, Germany, Springer (2004)
2. Barth, A., Franke, U.: Where will the oncoming vehicle be the next second? In: Intelligent Vehicles Symposium, IEEE. (2008)

3. Rosenhahn, B., Brox, T., Weickert, J.: Three-dimensional shape knowledge for joint image segmentation and pose tracking. International Journal of Computer Vision **73** (2007) 243–262
4. Wedel, A., Pock, T., Braun, J., Franke, U., Cremers, D.: Duality tv-l1 flow with fundamental matrix prior. In: Image and Vision Computing, Auckland, New Zealand (2008)
5. Lucas, B.D., Kanade, T.: An iterative image registration technique with an application to stereo vision. In: Proc. Seventh International Joint Conference on Artificial Intelligence, Vancouver, Canada (1981)
6. Rabe, C., Franke, U., Gehrig, S.: Fast detection of moving objects in complex scenarios. In: Intelligent Vehicles, Istanbul, Turkey, DaimerChrysler AG (2007) 398–403
7. Vedula, S., Baker, S., Rander, P., Collins, R., Kanade, T.: Three-dimensional scene flow. In: International Conference on Computer Vision. (1999)
8. Wedel, A., Rabe, C., Vaudrey, T., Brox, T., Franke, U., Cremers, D.: Efficient dense scene flow from sparce or dense stereo data. In: Proc. European Conf. on Computer Vision (ECCV). (2008)
9. Horn, B.K.P., Schunk, B.G.: Determining optical flow. In: Artificial Intelligence. Volume 17. (1981) 185–203
10. Memin, E., Perez, P.: Dense estimation and object-based segmentation of the optical flow with robust techniques. IEEE Transactions on Image Processing **7** (1998) 703–719
11. Brox, T., Bruhn, A., Papenberg, N., Weickert, J.: High accuracy optical flow estimation based on a theory for warping. In: European Conference on Computer Vision, Prague, Czech Republic (2004) 25–36
12. Bruhn, A., Weickert, J.: Towards ultimate motion estimation: Combining highest accuracy with real-time performance. Proc. Tenth International Conference on Computer Vision **1** (2005) 749–755
13. Zach, C., Pock, T., Bischof, H.: A duality based approach for realtime tv-l1 optical flow. In: DAGM. (2007)
14. Patras, I., Hendriks, E., Tziritas, G.: 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 of Computing and Imaging, Heijen, The Netherlands (1997)
15. Kalman, R.E.: A new approach to linear filtering and prediction problems. ASME-Journal of Basic Engineering **82** (1960) 35–45
16. Franke, U., Rabe, C.: Kalman filter based depth from motion with fast convergence. In: Proc. of the 2005 IEEE Intelligent Vehicles Symposium. (2005)
17. Franke, U., Rabe, C., Badino, H., Gehrig, S.: 6d-vision: Fusion of stereo and motion for robust environment perception. In: DAGM Symposium. (2005)
18. Wedel, A., Cremers, D., Pock, T., Bischof, H.: Structure- and motion-adaptive regularization for high accuracy optical flow. In: International Conference on Computer Vision. (2009)
19. Chambolle, A.: An algorithm for total variation minimization and applications. Journal of Mathematical Imaging and Vision **20** (2004) 89–97
20. Hirschmüller, H.: Accurate and efficient stereo processing by semi-global matching and mutual information. In: CVPR. (2005)
21. Bierman, G.J.: Factorization Methods for Discrete Sequential Estimation. Academic Press, Inc., New York (1977)
22. Rabe, C., Vaudrey, T.: University of Auckland. enpeda. image sequence analysis test site (EISATS). http://www.mi.auckland.ac.nz/EISATS (2010)