1st International Conference on Computer Vision Theory and Applications (VISAPP)
Setúbal, Portugal. 25 - 28 February, 2006

STEREO VISION-BASED DETECTION OF MOVING OBJECTS UNDER STRONG CAMERA MOTION

Hernán Badino, Uwe Franke, Clemens Rabe, Stefan Gehrig
DaimlerChrysler AG
71059 Sindelfingen, Germany
{*hernan.badino, uwe.franke, clemens.rabe, stefan.gehrig*}*@daimlerchrysler.com*

Keywords: Stereo Vision, Kalman Filter, Tracking, Ego-Motion Estimation

Abstract: The visual perception of independent 3D motion from a moving observer is one of the most challenging tasks in computer vision. This paper presents a powerful fusion of depth and motion information for image sequences. For a large number of points, 3D position and 3D motion is simultaneously estimated by means of Kalman Filters. The necessary ego-motion is computed based on the points that are identified as static points. The result is a real-time system that is able to detect independently moving objects even if the own motion is far from planar. The input provided by this system is suited to be used by high-level perception systems in order to carry out cognitive processes such as autonomous navigation or collision avoidance.

1 INTRODUCTION

The visual perception of independent 3D motion from a moving observer is one of the most challenging tasks in computer vision. Independent 3D motion is defined as the rigid or articulated change in position over time of an object with respect to an environment which is considered static. The perception of visual motion in animals and humans serves a wide variety of crucial roles: "way-finding (optic flow), perception of shape from motion, depth segregation, judgments of coincidence (time to collision, time to filling a tea cup), *judgments of motion direction and speed*, and perception of animate, biological activity" (Sekuler et al., 2004). In this paper we present a passive approach for the simultaneous estimation of position and velocity of single points, providing low-level information to more complex visual perception systems, such as autonomous navigation or collision avoidance. Treating such a process as a low-level task should not be surprising since, from the biological point of view, motion detection is a direct experience uniquely specified by the visual system.[1]

The further segmentation and integration of the information provided by our approach is here referred to as high-level vision, which must introduce some additional knowledge and intelligence to carry out a cognitive process ("a bicyclist is approaching from the left and we are going to collide with him within the next two seconds").

In order to estimate the velocity of a world point, we must be able to observe its change of position over time. The point position is easily obtained with multi-ocular platforms[2], which allow the instantaneous extraction of 3D position through triangulation. The time component is obtained by finding correspondences in consecutive frames. The correspondences are found between image points, i.e. optical flow or normal flow, or at the level of objects, which requires the previous segmentation of stereo points and the further tracking of objects. This last approach is commonly solved by an orthographical projection of the 3D points into an evidence-grid-like structure giving a bird-view of the scene, and grouping the projections according to their vicinity (Martin and Moravec, 1996). This method has its difficulties in segmenting distant objects and in separating distinct objects which are close together. The second op-

[1] Neurons in the middle temporal visual area integrate motion signals over large regions of visual space and respond to motion in their preferred direction, e.g. these neurons register motion information *per se*. More advanced perception activities are distributed over many areas of the brain, each extracting somewhat different information from the retinal image (Sekuler et al., 2004).

[2] For the state-of-the-art on monocular methods see for example (Kang et al., 2005), (Vidal, 2005) and (Woelk, 2004)

tion, i.e. normal flow, has some advantages with respect to optical flow in the sense that it reduces the correspondence problem (see for example (Argyros and Orphanoudakis, 1997) and (Morency and Darrell, 2002)). Nevertheless, normal flow is less informative compared to optical flow since it reflects only the motion in the direction of the image gradient and, therefore, does not provide an accurate estimation of point motion.

Methods based on optical flow have been widely proposed. One of the first attempts to fuse stereo and optical flow information was studied by Waxman and Duncan in (Waxman and Duncan, 1986), exploiting the relationship between 3D motion and image velocities with stereo constraints. Kellman and Kaiser (Kellman and Kaiser, 1995), Heinrich (Heinrich, 2002) and Mills (Mills, 1997) also make use of such constraints to detect independent motion. Demirdjian and Horaud (Demirdjian and Horaud, 2000) propose a method for the estimation of the ego-motion and the segmentation of moving objects. Demirdjian and Darrel (Demirdjian and Darrel, 2001) estimate rigid motion transformation mapping two reconstructions of a rigid scene in the disparity space (which they called d-motion). Dang *et al* (Dang et al., 2002) fuse optical flow and stereo disparity using Kalman Filters for object tracking, where the detection and segmentation of the object must have already been carried out. Kalman Filters were also used by Zhang and Faugeras (Zhang and Faugeras, 1991) for multiple motion estimation. Algorithms for the detection of moving objects using dense stereo and optical flow were proposed by Talukder and Matthies (Talukder and Matthies, 2004) and by Agrawal *et al* (Agrawal et al., 2005).

In this paper we describe an elegant approach for the estimation and continuous refinement of the position and velocity of world points. The six-dimensional state (i.e. 3D position and 3D velocity) as well as the six degrees of freedom (d.o.f.) of the ego-motion (translation and rotation in 3D Euclidian space) are estimated only based on the analysis of the images provided by the cameras and the required calibration parameters. Steps towards this approach have been described in (Franke et al., 2005) where ego-motion was restricted to a planar motion and obtained with the inertial sensors of the vehicle. Our approach combines binocular disparity and optical flow using Kalman Filters (KF), providing an iterative refinement of the 3D position and the 3D velocity of single points. Ego-motion estimation is achieved by computing the optimal rotation and translation between the tracked static points of multiple frames. These two processes are explained in detail in sections 2 and 3. In section 4 the whole algorithm is summarized. In Section 5 we present experimental results with real image sequences. In the last section we summarize the paper.

2 ESTIMATION OF 3D POSITION AND 3D VELOCITY

The main goal of our approach is the estimation of position and velocity of world points in 3D Euclidian space, and the recursively improvement of these estimates over time. A continuous improvement of the estimations is motivated by the noisy nature of the measurements and, therefore, Kalman Filters are the appropriate method to address this problem. In this section we describe a model which estimates the relative motion of world points relative to the observer, compensating the ego-motion of the camera platform with the observed motion of the points. If a world point is static, its observed motion is described as the inverse of the camera motion. Otherwise the point presents an independent 3D motion which we estimate. In the following subsections we assume a vehicle-based coordinate system, i.e. the origin of the coordinate system moves along with the observer.

2.1 Kalman Filter Model

Let suppose $\vec{x'}_{k-1} = (X, Y, Z)^T$ represents a world point observed by the system at time t_{k-1} and $\vec{v'}_{k-1} = (\dot{X}, \dot{Y}, \dot{Z})^T$ is its associated velocity vector. As the camera platform moves in its environment, so also does $\vec{x'}$ and after a time $\Delta\mathrm{t}_k$ the new position of the point in the vehicle coordinate system is given by:

$$\vec{x'}_k = \hat{R}_k \vec{x'}_{k-1} + \vec{\hat{d}}_k + \Delta\mathrm{t}_k \hat{R}_k \vec{v'}_{k-1} \qquad (1)$$

where $\hat{R}_k$ and $\vec{\hat{d}}_k$, are the rotation matrix and translation vector of the scene, i.e. the inverse motion of the camera. The velocity vector $\vec{v'}_k$, in vehicle coordinate system, changes its direction according to:

$$\vec{v'}_k = \hat{R}_k \vec{v'}_{k-1} \qquad (2)$$

Combining position and velocity in the state vector $\vec{x}_k = (X, Y, Z, \dot{X}, \dot{Y}, \dot{Z})^T$ leads to the discrete linear system model equation:

$$\vec{x}_k = A_k \vec{x}_{k-1} + B_k + \vec{\omega} \qquad (3)$$

with the state transition matrix

$$A_k = \begin{bmatrix} \hat{R}_k & \Delta\mathrm{t}_k \hat{R}_k \\ 0 & \hat{R}_k \end{bmatrix} \qquad (4)$$

and the control matrix

$$B_k = \begin{bmatrix} \vec{\hat{d}}_k \\ 0 \\ 0 \\ 0 \end{bmatrix} \qquad (5)$$

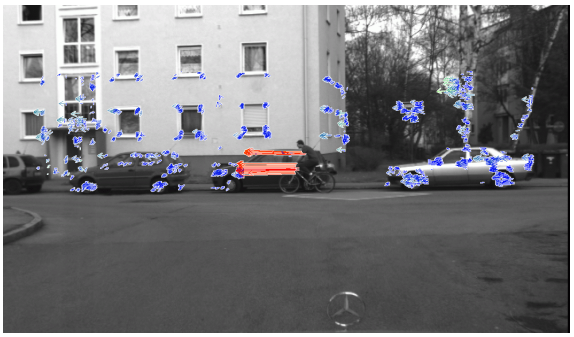


Figure 1: 3D motion vectors obtained when the vehicle drives on a flat street. The inertial sensors of the vehicle are used for the computation of the camera motion.

Figure 2: 3D motion vectors obtained when the vehicle drives on an uneven street. Only yaw-rate and velocity are considered in the ego-motion compensation.

The noise term $\vec{\omega}$ is assumed to be Gaussian white noise with a covariance matrix $\mathcal{Q}$.

The measurement model captures the information delivered by the stereo and optical flow systems. Assuming a pin-hole camera at the position $(0, height, 0)^T$ in the vehicle coordinate system with the optical axis parallel to the Z component, the non-linear measurement equation for a point given in the camera coordinate system is

$$z = \begin{bmatrix} u \\ v \\ d \end{bmatrix} = \frac{1}{Z} \begin{bmatrix} Xf_u \\ Yf_v \\ bf_u \end{bmatrix} + \vec{\nu} \qquad (6)$$

where (u, v) corresponds to the updated position of the point obtained with the optical flow, d is the disparity measured with stereo and (f_u, f_v) are the focal length in pixel width and height respectively. The noise term $\vec{\nu}$ is assumed to be Gaussian white noise with a covariance matrix $\mathcal{R}$.

This fusion of binocular disparity and optical flow offers a powerful tool for the iterative estimation of the position and velocity of world points. An example is shown in figure 1. The arrows correspond to the estimated direction. The warmth of the color encodes the 3D Euclidian velocity. Notice that the bicyclist and the vehicle at the background are close together. Nevertheless a clear segmentation of both is possible thanks to the additional information of velocity. The rotation matrix and the translation vector of equations 1 to 5 are normally obtained from the inertial sensors of the vehicle which usually consist of a speedometer and a yaw-rate sensor (Franke et al., 2005). For indoor applications the information delivered by these inertial sensors is usually sufficient in order to robustly estimate the position and velocity of world points. However, for many other applications (e.g. vehicle assistance systems) the motion of the camera platform is not purely planar. Figure 2 shows an example when the cameras are mounted in a vehicle. It is obvious that the error in the velocity vectors is produced by the roll rotation. A better approach is possible if all six d.o.f. are considered computing the ego-motion only from the images.

3 ROBUST EGO-MOTION ESTIMATION

Computing ego-motion from an image sequence means obtaining the change of position and orientation of the observer with respect to a static scene, i.e. the motion is relative to an environment which is considered static. In most approaches this fact is exploited and ego-motion is computed as the inverse of the scene motion (Demirdjian and Horaud, 2000), (Mallet et al., 2000), (Matthies and Shafer, 1987), (Olson et al., 2003), (van der M. et al., 2002), (Badino, 2004). In the latter a robust approach for the accurate estimation of the six d.o.f. of motion (three components for translation and three for rotation) in traffic situations is presented. In this approach, stereo is computed at different times and clouds of 3D points are obtained. The optical flow establishes the point-to-point correspondence. The motion of the camera is computed with a least-squares approach finding the optimal rotation and translation between the clouds. In order to avoid outliers, a smoothness motion constraint is applied rejecting all correspondences which are inconsistent with the current motion. This last two steps are also applied between non-consecutive frames. In the next sub-sections we briefly review the main steps of this approach.

3.1 Obtaining the Absolute Orientation Between Two Frames

Let $X = \{\vec{x}_i\}$ be the set of 3D points of the previous frame and $P = \{\vec{p}_i\}$ the set of 3D points observed at the current frame, where $\vec{x}_i \leftrightarrow \vec{p}_i$, i.e. $\vec{p}_i$ is the transformed version at time t_k of the point $\vec{x}_i$ at

time t_{k-1}. In order to obtain the motion of the camera between the current and the previous frame we minimize a function which is expressed as the sum of the weighted residual errors between the rotated and translated data set X with the data set P, i.e.:

$$\sum_{i=1}^{n} w_i \|\vec{p}_i - R_k \vec{x}_i - \vec{d}_k\|^2 \quad (7)$$

where n is the amount of points in the sets, R_k is a rotation matrix, $\vec{d}_k$ is a translation vector, and w_i are individual weights representing the expected error in the measurement of the points. To solve this least-squares problem we use the method presented by Horn (Horn, 1987), which provides a closed form solution using unit quaternions. In this method the optimal rotation quaternion is obtained as the eigenvector corresponding to the largest positive eigenvalue of a 4×4 matrix. The quaternion is then converted to the rotation matrix. The translation is computed as the difference of the centroid of data set P and the rotated centroid of data set X. The computation of the relative orientation is not constrained to this specific method. Lorusso *et al* (Lorusso et al., 1995) shortly describe and compare this and another three methods for solving this problem in closed form.

3.2 Motion Representation with Matrices

In order to simplify the notation of the following subsections, we represent the motion in homogeneous coordinates. The computed motion of the camera between two consecutive frames, i.e. from frame $k-1$ to frame k, is represented by the matrix M'_k where:

$$M'_k = \begin{bmatrix} R_k & \vec{d}_k \\ 0 & 1 \end{bmatrix} \quad (8)$$

The rotation matrix $\hat{R}_k$ and translation vector $\vec{\hat{d}}_k$ from equations 1 to 5 are obtained by just inverting M'_k, i.e.:

$$M_k^{'-1} = \begin{bmatrix} \hat{R}_k & \vec{\hat{d}}_k \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} R_k^{-1} & -R_k^{-1}\vec{d}_k \\ 0 & 1 \end{bmatrix} \quad (9)$$

The total motion of the camera since initialization can be obtained as the products of the individual motion matrices:

$$M_k = \prod_{i=1}^{k} M'_i \quad (10)$$

A sub-chain of movements from time t_n to time t_m is:

$$M_{n,m} = M_n^{-1} M_m = \prod_{i=n+1}^{m} M'_i \quad (11)$$

Figure 3 shows an example of motion integration with matrices. As we will show later in section 3.4 equation 11 will support the integration of the motion between two non-consecutive frames (multi-step estimation).

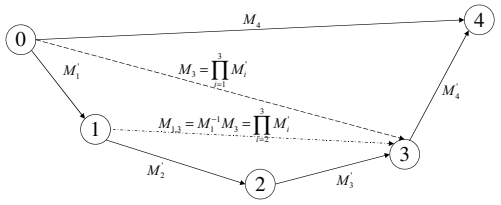


Figure 3: The integration of single-step estimations can be obtained by just multiplying the individual motion matrices. Every circle denotes the state (position and orientation) of the camera between time t_0 and time t_4. Vectors indicate motion in 3D-space.

3.3 Smoothness Motion Constraint

Optical flow and/or stereo can deliver false information about 3D position or image point correspondence between image frames. Some of the points might also correspond to an independently moving object. A robust method should still be able to give accurate results facing such situations. If the frame rate is high enough in order to obtain a smooth motion between consecutive frames, then the current motion is similar to the immediate previous motion. Therefore, before including the pair of points $\vec{p}_i$ and $\vec{x}_i$ into their corresponding data sets P and X, we evaluate if the vector $\vec{v}_i = \overrightarrow{p_i x_i}$ indicates a coherent movement. Let us define $\vec{m} = [\ \dot{x}_{max} \quad \dot{y}_{max} \quad \dot{z}_{max} \quad 1\ ]$ as the maximal accepted error of the position of a 3D point with respect to a predicted position. Based on our previous ego-motion estimation step we evaluate the motion coherence of the vector $\vec{v}_i$ as:

$$\vec{c}_i = M'_{k-1}\vec{x}_i - \vec{p}_i \quad (12)$$

i.e. the error of our prediction. If the absolute value of any component of c_i is larger than $\vec{m}$ the pair of points are discarded and not included in the data sets for the posterior computation of relative orientation. Otherwise we weight the pair of points as the ratio of change with respect to the last motion:

$$w_i = 1 - \frac{\|\vec{c}_i\|^2}{\|\vec{m}'\|^2} \quad (13)$$

which is later used in equation 7. Equations 12 and 13 define the smoothness motion constraint (SMC).

3.4 Multi-Frame Estimation

Single step estimation, i.e. the estimation of the motion parameters from the current and previous frame

is the standard case in most approaches. If we are able to track points over m frames, then we can also compute the motion between the current and the m previous frames and integrate this motion into the single step estimation (see figure 4). The estimation of motion between frame m and the current frame k ($m < k-1$) follows exactly the same procedure as explained above. Only when applying the SMC, a small change takes place, since the prediction of the position for $k - m$ frames is not the same as for a single step. In other words, the matrix M'_{k-1} of equation 12 is not valid any more. If the single step estimation for the current frame was already computed as $\tilde{M}_k$ equation 12 becomes:

$$\vec{c_i} = M_{k-m}^{-1} M_{k-1} \tilde{M}_k \vec{x}_i - \vec{p}_i. \qquad (14)$$

Equation 14 represents the estimated motion between times t_{k-m} and t_{k-1} (from equation 11), updated with the current simple step estimation of time t_k. This allows the SMC to be even more precise, since the uncertainty in the movement is now based on an updated prediction. On the contrary in the single step estimation, the uncertainty is based on a position defined by the last motion.

Once the camera motion matrix $\tilde{M}_{m,k}$ between times t_{k-m} and t_k is obtained, it is integrated with the single step estimation. This is performed by an interpolation. The interpolation of motion matrices makes sense if they are estimations of the same motion. This is not the case since the single step motion matrix is referred to as the motion between the last two frames and the multi-step motion matrix as the motion between m frames in the past to the current one. Thus, the matrices to be interpolated are $\tilde{M}_k$ and $M_{m,k-1}^{-1} \tilde{M}_{m,k}$ (see figure 4). The corresponding rotation matrices are converted to quaternions in order to apply a spherical linear interpolation. The interpolated quaternion is converted to the final rotation matrix R_k. Translation vectors are linearly interpolated, obtaining the new translation vector $\vec{t}_k$. The factors of the interpolation are given by the weighted sum of the quadratic deviations obtained when computing the relative motion of equation 7.

The multi-frame approach performs better thanks to the integration of more measurements. It also reduces the integration of the errors produced by the single-step estimation between the considered time points. In fact, our experience has shown that without the multi-frame approach the estimation degenerates quickly and, normally, after a few hundred frames the ego-position diverges dramatically from the true solution. Thus, the multi-frame approach provides additional stability to the estimation process.

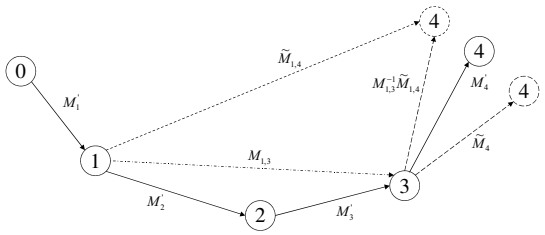


Figure 4: multi-frame approach. Circles represent the position and orientation of the camera. Vectors indicate motion in 3D-space. $\tilde{M}_4$ (single step estimation) and $M_{1,3}^{-1}\tilde{M}_{1,4}$ (multi-step estimation) are interpolated in order to obtain the final estimation M'_4.

4 THE ALGORITHM

This section summarizes the main tasks of our approach. *1. Left and Right Image Acquisition.*

2. Measurement.
- *a. Compute optical flow (tracking).*
- *b. Compute stereo disparity.*

3. Ego-Motion Computation.
- *a. Apply SMC Single Step.*
- *b. Compute Single Step (SS) estimation.*
- *c. Apply SMC Multi-Step.*
- *d. Compute Multi-Step (MS) estimation.*
- *e. Interpolate SS and MS results.*

4. Kalman Filter.
- *a. Compute A and B matrices with ego-motion estimation (Step 3).*
- *b. Update models.*

5. Go to Step 1.

5 EXPERIMENTAL RESULTS

Our current implementation tracks 1200 points. It runs at $12 - 16$ Hz on a 3.2 GHz PC. We use a speed-optimized version of the KLT algorithm (Shi and Tomasi, 1994) and compute stereo using a coarse to fine correlation method as described in (Franke, 2000). In order to show the robustness of the method, we demonstrate the performance on three real world sequences.

Uneven Street. In this sequence, the vehicle drives on a slightly uneven street. A bicyclist appears suddenly from a back street at the right. The sequence has 200 stereo image pairs and was taken at 16 Hz. The baseline of the stereo system is 0.35 meters and the images have a VGA resolution. The car starts from a standing position and accelerates up to a velocity of 30 km/h.

Figure 5(a) shows the improved results for the estimation already illustrated in figure 2. Thanks to the ego-motion estimation, the artefacts disappear and the bicyclist is clearly visible now. Figure 5(b) shows the

ego-motion estimation results for every frame of the sequence. The minimum in the roll rotation at frame 120 correspond to the same frame number as above. Notice that the lateral translation decreases together with the velocity despite the small yaw rotation. This indicates that the optical axis of the camera is not aligned with the driving direction of the vehicle, i.e. the camera is fixed mounted on the vehicle but viewing slightly to the right.

This last observation reveals another advantage of computing the ego-motion directly from the images: there is no need of an external calibration of the camera platform w.r.t. the vehicle. Figure 6(a) shows the velocity vectors for frame 146, using only inertial sensors when the yaw rotation of the camera w.r.t. the vehicle was wrongly estimated by 5 degrees. Figure 6(b) shows the results for the same camera configuration using only ego-motion estimation.

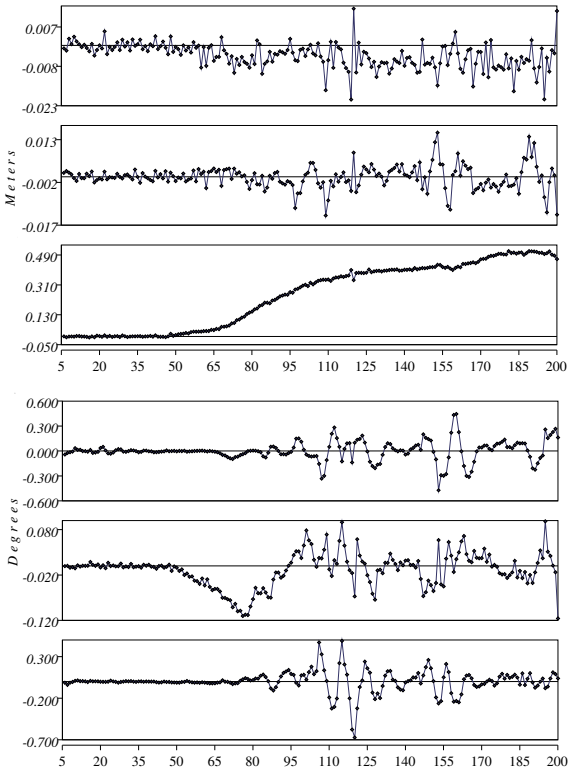
Crossing Traffic with Background Occlusion. Figure 7 shows the estimation results with an oncoming bus. In this sequence, both the ego-vehicle and the bus are driving in a curve. In the worst case, the bus takes up more than 35% of the image area. The computation of ego-motion is still possible here thanks to the smoothness motion constraint, which selects only static points (blue points in the image) for the ego-motion computation.

Indoor Environment. Figure 8 shows an example of the results in an indoor environment. The sequence shows a person walking behind a desktop while the camera is translated and rotated by hand. The color encoding is the same as in the two previous examples but red now means 1.75 m/s. The different colors of the arrows on the moving person correspond to the different velocities of the different body parts (arms, body and legs).

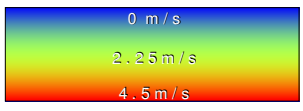
6 SUMMARY

We have presented a method for the efficient estimation of position and velocity of tracked image points, and for the iterative improvements of these estimates over time. The scene structure is obtained with a binocular platform. Optical flow delivers the change of this structure in consecutive frames. By observing the dynamics of the individual 3D points, an estimation of their velocity is possible. This way we obtain a six dimensional state for every world point which is given in a vehicle coordinate system. As the camera moves, the state of every point needs to be updated according to the motion of the observer and, therefore, the ego-motion of the cameras needs to be known. Instead of obtaining motion information from the error-prone inertial sensors of the vehicle, the six d.o.f. of ego-motion are obtained as the optimal rotation and translation between the current and previous set of static 3D points of the environment. All this information is fused using Kalman Filters, obtaining

(a) 3D motion vectors. The arrows correspond to the estimated 3D position of the points in 0.5 seconds back projected into the image. The color encodes the velocity as shown in image (c)

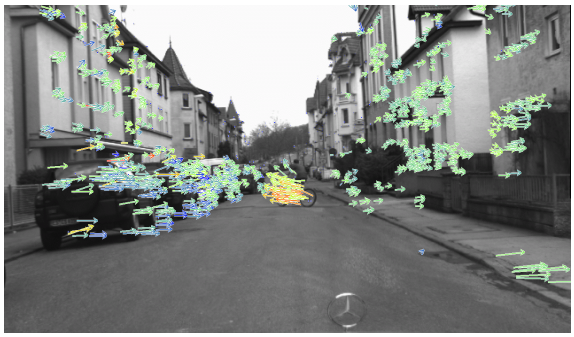


(b) Ego-motion estimation. The plots show the observed motion between consecutive frames. From top to bottom: X, Y, Z translations; pitch, yaw, roll rotations.



(c)

Figure 5: a bicyclist approaches from the right while the vehicle drives on an uneven street.



(a) 3D motion vectors using inertial sensors.

(b) 3D motion vectors estimating the six d.o.f. of motion.

Figure 6: results when the camera orientation was wrong estimated. The images shows the state of the scene 1.5 seconds later as in figure 5.

an iterative improvement over time of the 3D point position and velocity.

This algorithm turned out to be powerful for the detection of moving obstacles in traffic scenes. However, it is not limited to this application but may be also useful for other applications such as autonomous navigation, simultaneous localization and mapping or target detection for military purposes.

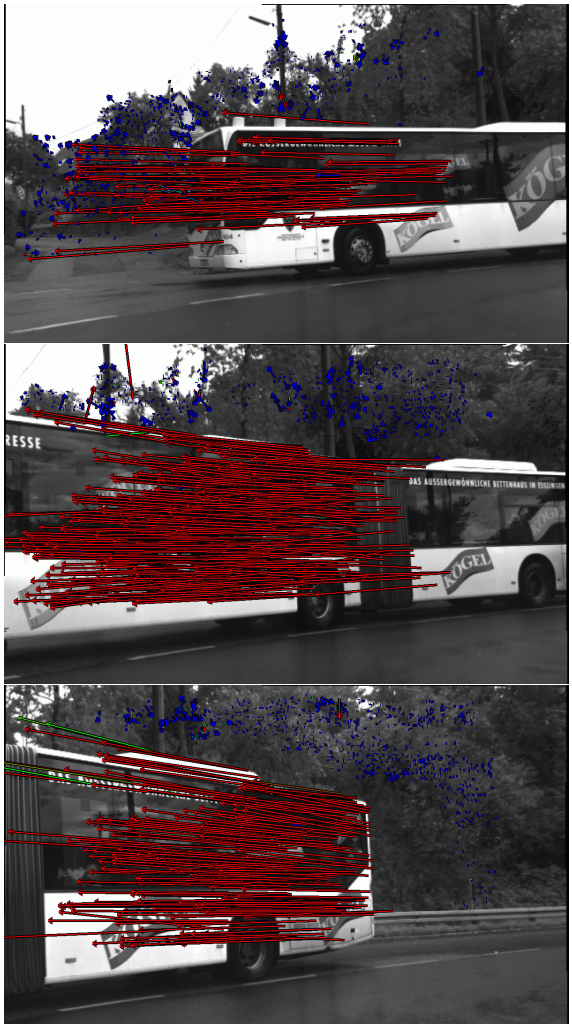


Figure 7: estimation results when the moving objects takes up a large area of the image. The time between the images is 0.5 seconds.

REFERENCES

Agrawal, M., Konolige, K., and Iocchi, L. (2005). Real-time detection of independent motion using stereo. In *IEEE Workshop on Motion and Video Computing (WACV/MOTION 2005)*, pages 672–677, Breckenridge, CO, USA.

Argyros, A. A. and Orphanoudakis, S. C. (1997). Independent 3d motion detection based on depth elimination in normal flow fields. In *Proceedings of the 1997 Conference on Computer Vision and Pattern Recognition (CVPR '97)*, pages 672–677.

Badino, H. (2004). A robust approach for ego-motion estimation using a mobile stereo platform. In 1^{st} *International Workshop on Complex Motion (IWCM'04)*, Günzgburg, Germany. Springer.

Dang, T., Hoffmann, C., and Stiller, C. (2002). Fusing optical flow and stereo disparity for object tracking. In *Proceedings of the IEEE V. International Conference on Intelligent Transportation Systems*, pages 112–117, Singapore.

Demirdjian, D. and Darrel, T. (2001). Motion estimation from disparity images. In *Technical Report AI Memo 2001-009, MIT Artificial Intelligence Laboratory*.

Demirdjian, D. and Horaud, R. (2000). Motion-egomotion discrimination and motion segmentation from image pair streams. In *Computer Vision and Image Understanding*, volume 78(1), pages 53–68.

Franke, U. (2000). Real-time stereo vision for urban traffic scene understanding. In *IEEE Conference on Intelligent Vehicles*, Dearborn.

Franke, U., Rabe, C., Badino, H., and Gehrig, S. (2005). 6d-vision: Fusion of stereo and motion for robust environment perception. In *DAGM '05*, Vienna.

Heinrich, S. (2002). Fast obstacle detection using

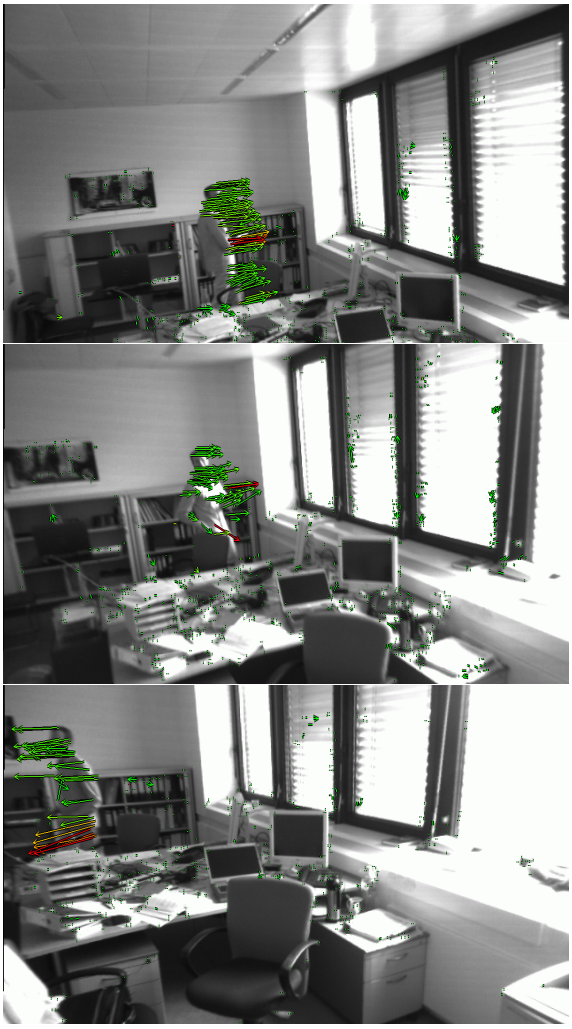


Figure 8: a person moves first to the right (first image), turn around (second image) and finally walk to the left (third image) while the camera is translated and rotated with the hand.

flow/detph constraint. In *IEEE Intelligent Vehicle Symposium (IV'2002)*, Versailles, France.

Horn, B. K. P. (1987). Closed-form solution of absolute orientation using unit quaternions. In *Journal of the Optical Society of America A*, volume 4(4), pages 629–642.

Kang, J., Cohen, I., Medioni, G., and Yuan, C. (2005). Detection and tracking of moving objects from a moving platform in presence of strong parallax. In *IEEE International Conference on Computer Vision ICCV'05*.

Kellman, P. J. and Kaiser, M. (1995). Extracting object motion during observer motion: Combining constraints from optic flow and binocular disparity. *JOSA-A*, 12(3):623–625.

Lorusso, A., Eggert, D., and Fisher, R. B. (1995). A comparison of four algorithms for estimating 3-d rigid transformations. In *Proc. British Machine Vision Conference*, Birmingham.

Mallet, A., Lacroix, S., and Gallo, L. (2000). Position estimation in outdoor environments using pixel tracking and stereovision. In *Proceedings of the 2000 IEEE International Conference on Robotics and Automation*, pages 3519–3524, San Francisco.

Martin, M. and Moravec, H. (1996). Robot evidence grids. *The Robotics Institute Carnegie Mellon University (CMU-RI-TR-96-06)*.

Matthies, L. and Shafer, S. A. (1987). Error modeling in stereo navigation. In *IEEE Journal of Robotics and Automation*, volume RA-3(3), pages 239–248.

Mills, S. (1997). Stereo-motion analysis of image sequences. In *Proceedings of the first joint Australia and New Zealand conference on Digital Image and Vision Computing: Techniques and Applications, DICTA'97 / IVCNZ'97, Albany, Auckland, NZ.*

Morency, L. P. and Darrell, T. (2002). Stereo tracking using icp and normal flow constraint. In *Proceedings of International Conference on Pattern Recognition*, Quebec.

Olson, C. F., Matthies, L. H., Schoppers, M., and Maimone, M. W. (2003). Rover navigation using stereo ego-motion. In *Robotics and Autonomous Systems*, volume 43(4), pages 215–229.

Sekuler, R., Watamaniuk, S. N. J., and Blake, R. (2004). *Stevens' Handbook of Experimental Psychology*, volume 1, Sensation and Perception, chapter 4, Perception of Visual Motion. Wiley, 3^{rd} edition.

Shi, J. and Tomasi, C. (1994). Good features to track. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR'94)*.

Talukder, A. and Matthies, L. (2004). Real-time detection of moving objects from moving vehicles using dense stereo and optical flow. In *Proceedings of the IEEE International Conference on Intelligent Robots and Systems*, pages 315–320, Sendai, Japan.

van der M., W., Fontijne, D., Dorst, L., and Groen, F. C. A. (2002). Vehicle ego-motion estimation with geometric algebra. In *Proceedings IEEE Intelligent Vehicle Symposium, Versialles, France, May 18-20 2002*.

Vidal, R. (2005). Multi-subspace methods for motion segmentation from affine, perspective and central panoramic cameras. *IEEE International Conference on Robotis and Automation*.

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

Woelk, F. (2004). Robust monocular detection of independent motion by a moving observer. In 1^{st} *International Workshop on Complex Motion (IWCM'04)*, Günzgburg, Germany. Springer.

Zhang, Z. and Faugeras, O. D. (1991). Three-dimensional motion computation and object segmentation in a long sequence of stereo images. Technical Report RR-1438, Inria.