Extrinsic Camera Calibration in Vehicles with Explicit Ground Estimation

Frank Pagel and Dieter Willersinn
Fraunhofer Institute of Optronics, System Technologies and Image Exploitation (IOSB)
Autonomous Systems and Machine Vision
76131 Karlsruhe, Germany
{frank.pagel, dieter.willersinn}@iosb.fraunhofer.de

***Abstract*— The use of multiple cameras in vehicles becomes more and more attractive as hardware prices decrease rapidly. Multiple camera sensors can be used to cover larger areas of the environment of a vehicle and for 3D scene reconstruction using stereo or structure from motion techniques. To be able to merge the sensor data in a common coordinate frame, it is necessary to know the relative positions and orientations of the cameras. However, as camera configurations may have non-overlapping fields of view due to cost or design reasons, no point correspondences between the camera images can be used. Instead, we apply a motion-based technique that allows general, in especially non-overlapping camera configurations. By estimating the ground parameters in an intermediate step, we overcome the typical problems of purely planar motion. Finally we are able to estimate all 6 Euclidean calibration parameters between each camera pair.**

This contribution outlines a concept to perform an online calibration of multiple cameras on a mobile platform with non-overlapping fields of view. Results with simulated data are presented.

***Index Terms*— Extrinsic Calibration, Online Calibration, Non-overlapping Views.**

I. Introduction

More and more cameras are mounted on mobile robots to improve their situation awareness. Cameras are also in use in modern advanced driver assistant systems to provide additional environmental information or to give warnings in critical situations to the driver. Structure from motion techniques (SFM) are widely used to reconstruct the three-dimensional structure of the surrounding scene. This contribution focuses on a multiocular camera configuration on a moving platform with sparsely or non-overlapping fields of view (FOV).

3D scene reconstruction with multiple cameras is a growing field of research [1], [4]. Considering modern SFM techniques, single cameras can also be used for egomotion estimation [5]. Most important for all multiocular reconstruction tasks is the knowledge of the camera parameters, that are the intrinsic (focal length, principal point and lens distortion) and the extrinsic (rotation and translation) parameters. Extrinsic parameters describe the geometric relationship between the cameras. Common calibration techniques fail because of the non-overlapping FOV. The cameras do not see the same scene and hence no corresponding image features can be used. This paper adresses the online calibration without using any pattern or known scene structure.

II. Related Work and Proposed Solution

Lamprecht et al. [3] use well-known patterns from the scene, e. g. traffic signs, to determine the extrinsic parameters of two non-overlapping cameras in a vehicle. The drawback of this approach is that the localization of the pattern must be very precise to perform a precise calibration. Once a pattern is detected in a camera it must be stored and redetected in the other cameras where the object may occur projectively distorted or from a completely different view.

A purely motion-based approach was proposed by Esquivel et al. [2]. They only use the trajectories of the single cameras to determine the extrinsic configuration. Similar to [3] this approach needs a robust egomotion estimation and could be used offline as well as online. Unfortunately, the algorithm strongly depends on the rotational motion and hence the quality of the calibration suffers from the almost planar motion of regular ground vehicles.

Ruland et al. [7] proposed an extrinsic calibration method for a two camera system with non-overlapping views and fixed camera height. So the

problem of planar motion was solved by simply fixing the longitudinal translation parameter. However they did not present a scalable solution for multi camera applications.

Pagel et al. [6] proposed a general, scalable solution of calibrating multi camera rigs with non-overlapping views by propagating and merging the motion and calibration parameters. But they left the problem of planarly moving cameras still unsolved, too.

Our goal is to determine the extrinsic calibration parameters of a set of multiple cameras on a moving platform. The intrinsic parameters are assumed to be known. As the proposed calibration procedure is based on the cameras' motion, there is also need for an accurate motion estimation. Each camera is embedded into a module that serves as a calculation unit and hence manages the motion and calibration states. Furthermore we are interested in a global state estimation. *Global* in this context means, that the states of all other modules are considered when a single module's state is estimated. In contrast, when a module's state is calculated only based on the local sensor data, it is called *local*. All local estimations in this contribution are performed with an extended Kalman filter.

A global overall optimization for all calibration and motion parameters is likely to fail because of the large dimension of the resulting state vector. For N cameras there are N motion vectors and $N(N-1)/2$ extrinsic transformation parameters to estimate. As a Eucliden transformation can be described with three rotational and three translational parameters, such a global model would result in a $6 \cdot (N + N(N-1)/2)$-dimensional state vector.

The concept of propagating and merging spatial transformation parameters and its uncertainties was already used by Smith and Cheeseman [8]. This concept can be transferred to our calibration purposes. Instead of calculating the global state of a whole camera rig with a single Kalman filter that uses all sensor measurements simultaneously, the local state of each module is calculated first. Then, by using the initial extrinsic calibration and motion parameters, the local motions and extrinsic parameters as well as the corresponding errors can be propagated for the other modules. Afterwards the (local) propagations and uncertainties can be fused to get a global estimation. Such an approach is much more effective in practice because of its scalabilty and the lower computational cost per module. Our approach also ensures that the communication bandwidth between the modules is kept low.

The approach of local optimization, propagation and global fusion can be applied for both motion and extrinsic parameter estimation. Therefore a module's calibration process is devided into a motion and a calibration step.

The whole algorithm can be outlined as follows:

1) Local motion estimation
2) Local motion propagation (which is the local guess of the other motions)
3) Global motion estimation (which is the fusion of the propagated local motions)
4) Ground plane estimation
5) Local estimation of the $N-1$ transformations describing the position and orientation of the other modules in the ground plane
6) Local calibration propagation (which is the local guess of all remaining extrinsics)
7) Global fusion of the local calibration guesses
8) Merging the transformations to get the final 6 dof extrinsic Euclidean transformations

III. Geometric Model

In this Section we shortly present the parameters that are necessary to describe the complete geometric structure of a moving camera rig. Both the motion of a single camera and the relative position of two cameras can be considered as a Euclidean transformation. The transformation between two camera modules M_i and M_j at time t is given by the transformation matrix

$$T_{ij} = \begin{pmatrix} \mathbf{R}_{ij} & \mathbf{t}_{ij} \\ \mathbf{0}^T & 1 \end{pmatrix}_{4\times 4}.$$

$\mathbf{R}(r_x, r_y, r_z) \in \mathbb{R}^{3\times 3}$ is a rotation matrix with $\mathbf{R}^T\mathbf{R} = \mathbf{R}\mathbf{R}^T = \mathbf{I}$ and $\mathbf{t} \in \mathbb{R}^3$ is a translation vector. The motion of camera M_i between two time steps k and $k+1$ is given by

$$\Omega_i = \begin{pmatrix} \mathbf{W}_i & \mathbf{v}_i \\ \mathbf{0}^T & 1 \end{pmatrix}_{4\times 4},$$

with rotation matrix $\mathbf{W}(\omega_x, \omega_y, \omega_z) \in \mathbb{R}^{3\times 3}$ and translation vector $\mathbf{v} \in \mathbb{R}^3$.

The transformation into the ground plane G_i can be described with three parameters $t_{y_{g_i}}, r_{x_{g_i}}, r_{z_{g_i}}$. The two rotational parameters make the camera parallel to the plane and the translational component fits the camera into the plane. In the ground plane, the extrinsic transformation is given by

$$T'_{ij} = \begin{pmatrix} \mathbf{R}'_{ij} & \mathbf{t}'_{ij} \\ \mathbf{0}^T & 1 \end{pmatrix}_{4\times 4}$$

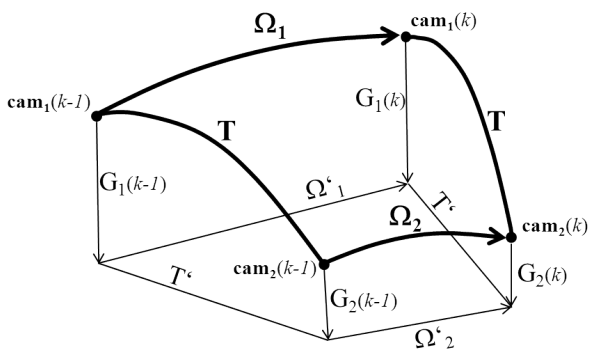

Fig. 1. Basic geometric constellation for a 2-camera rig. T is the extrinsic calibration matrix, Ω is the camera motion and G are the transformations into the ground plane. Transformations in the ground plane are labeled with a prime.

with $\mathbf{R}'(0, r'_y, 0)$, $\mathbf{t}' = (t'_x, 0, t'_z)^T$ and

$$T_{ij} = G_i(k) \cdot T'_{ij} \cdot G_j^{-1}(k). \tag{1}$$

The cameras' motions in the ground plane are then given by

$$\Omega'_i = G_i^{-1}(k-1) \cdot \Omega_i \cdot G_i(k) \tag{2}$$

with

$$\Omega'_i = \begin{pmatrix} \mathbf{W}'_i & \mathbf{v}'_i \\ \mathbf{0}^T & 1 \end{pmatrix}_{4\times 4}.$$

All these relations are shown in Fig. 1.

IV. Camera Motion

A. Local Motion Estimation

Motion or egomotion estimation purely based on camera data is also known as *visual odometry*. Here, a modified version of the motion estimation approach of Pagel [5] is used. The motion parameters of a single camera are estimated via a robust iterated extended Kalman filter (RIEKF) as proposed by Dang et al. [1]. As in [1], the motion parameters Ω_i of module M_i are determined by minimizing the epipolar constraint, the trifocal constraint and the projection error within the function

$$h_{mot}(\Omega_i, \mathbf{z}) \tag{3}$$

with respect to Ω_i and measurement $\mathbf{z} = (..., \mathbf{z}_i, ...)^T$. A single measurement is given by $\mathbf{z}_i = (\mathbf{v}, z)^T$, where $\mathbf{v}$ is an optical flow triple $\mathbf{v} = (\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3)^T$ and z is the depth of the respective scene point. z can be precalculated by triangulating $\mathbf{x}_1$ and $\mathbf{x}_2$ using the calculated motion of the previous time step. The robust preprocessing step detects measurement outliers in a RANSAC-like procedure. The RIEKF also refines the measurement during the iteration process.

The Kalman filter here has two big advantages: First, it can be easily extendended with other sensor sources (e. g. odometry data) or motion estimation approaches (e. g. fundamental matrix factorization) by adjoining them to the prediction step. And second and most important, the Kalman filter provides an uncertainty of the current estimation in form of a covariance matrix that is necessary for the propagation and fusion step.

B. Motion Propagation

After each camera module has estimated its own egomotion (Fig. 2a), we can now determine the motion parameters of all other modules by using the calibration parameters (Fig. 2b). Even in the initial case when the extrinsic parameters are still unknown, the uncertainty can be set extremely high so that only the local estimations fall into account. Finally, each module can calculate a local estimation of the global model by considering uncertainties of the calibration and egomotion estimations (Fig. 2c). The propagation step is described in detail in Pagel et al. [6].

C. Global Motion Fusion

From the local propagation we have an estimation of each camera motion from each of the N modules (Fig. 2d). These N estimations per motion are now merged to one. The fusion of the states $\mathbf{x}$ and covariances $\mathbf{\Sigma}$ can be done in a pairwise manner following the approach of Smith & Cheeseman [8]:

$$\mathbf{x}_F = \mathbf{x}_1 + \mathbf{\Sigma}_1 \cdot (\mathbf{\Sigma}_1 + \mathbf{\Sigma}_2)^{-1} \cdot (\mathbf{x}_2 - \mathbf{x}_1) \tag{4}$$

and

$$\mathbf{\Sigma}_F = \mathbf{\Sigma}_1 - \mathbf{\Sigma}_1 \cdot (\mathbf{\Sigma}_1 + \mathbf{\Sigma}_2)^{-1} \cdot \mathbf{\Sigma}_1. \tag{5}$$

V. Calibration

As is illustrated in Fig. 1 the extrinsic transformation and the motion transformations are related by

$$T_{ij} = \Omega_i^{-1} T_{ij} \Omega_j \tag{6}$$

which leads to

$$(\mathbf{I} - \mathbf{W}_i)\mathbf{t}_{ij} = \mathbf{v}_i - \mathbf{R}_i j \mathbf{v}_j. \tag{7}$$

To be able to determine $\mathbf{t}$ from this equation, some conditions must be fulfilled as was shown by

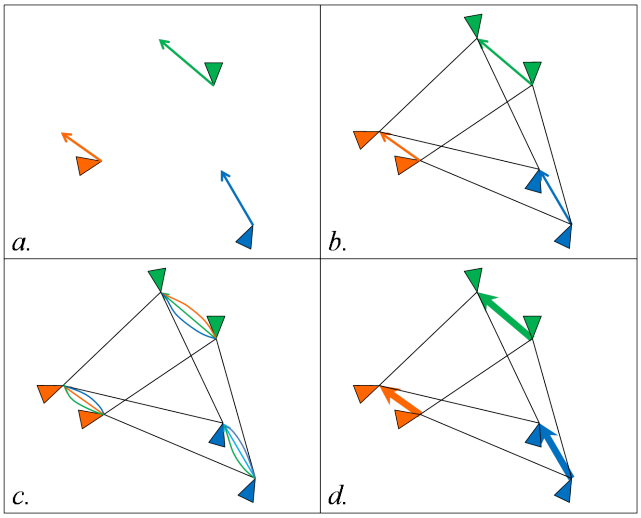

Fig. 2. *a.* Each modul performs a local motion estimation with a Kalman filter using the local sensor data. *b.* Extrinsic calibration parameters are taken into account. *c.* Each modul determines the motions of the other modules based on the extrinsic parameters via state and error propagation. Hence, there are N guesses for each of the N motions. *d.* The N local guesses of each module's motion are fused to a global estimation using the DKF approach.

Tsai & Lenz [9]. First of all, $\mathbf{W}_1 \neq \mathbf{I}$, otherwise $(\mathbf{I} - \mathbf{W}_1) = \mathbf{0}$. Furthermore, as $\mathbf{W}_1$ is a rotation matrix, we have $det(\mathbf{I} - \mathbf{W}_1) = 0$, so we need more than just one single motion to be able to estimate the translational component of the extrinsic parameters. And as was shown be Tsai & Lenz [9] all motions must not be coplanar.

This has some implications for our algorithm. As we want to run the calibration continously, we need to build up a history of motions. This is achieved by accumulating the last H motions from $\Omega(k)$ to $\Omega(k-h)$, $h = 1, ..., H$. Though the accumulation of motions lead to bigger rotational magnitudes, but also the estimation errors are propagated and can cause problems for the extrinsic estimation. So H should be chosen carefully. In our implementation we chose $H = 10$.

All motions with rotation angles $< \epsilon$ are not considered in the calibration step. The (mostly) coplanar motion of street vehicles affects that the longitudinal component of the translation vector cannot be determined. For that reason we perform an intermediate processing step where we estimate the transformations from the camera coordinate systems to the ground plane. Once these transformations are known, the extrinsic calibration problem can be solved in a 2D-plane and reduces to an estimation of only 3 parameters.

A. Ground Estimation

Once the motion parameters are known, we can estimate the 3D coordinates of the point correspondences we already used for the motion estimation in section IV-A. The transformation of a camera into a plane can be modeled by only three parameters: $t_{g_y}, r_{g_x}, r_{g_z}$. We estimate these parameters again with a RIEKF. Input data are 3D points calculated from the flow vectors and the cameras' motions. Again, the robust preprocessing is quite important, because only 3D points that lie in the ground plane should be considered. The Kalman filter estimates the three parameters by forcing the Y-component of the 3D points to be zero. With $t_{g_y}, r_{g_x}, r_{g_z}$ we can determine the homogeneous 4×4-transformations G_i at time k.

In usual traffic scenes there will always be scenes where no image points on the ground floor can be selcted (imagine a side looking camera closely passing a row of parking cars). Fortunately, there is no need to estimate the calibration parameters the whole time, as we assume the parameters to be fixed (the parameter changes of the extrinsic configuration is considered to be long-term changes). When no ground parameters can be estimated, the parameter prediction is just an identity function (the same principle works for the extrinsic parameters when the rotational motion is less than a given threshold).

B. Local Extrinsic Parameter Estimation in the Ground Plane

Once the motion parameters and the current and the previous ground transformation parameters are known, the problem of estimating the 6 dof extrinsic parameters is reduced to estimating 3 parameters t_x, t_z, r_y. These are the extrinsic parameters in the ground plane which describe the homogeneous transformation T'. Now we transform the motions Ω_i into the ground plane $\Omega'_i = G_j^{-1}(k-1) \cdot \Omega_j \cdot G_j(k)$. The parameter estimation is again implemented as an RIEKF. The measurements are the accumulated motions Ω'_i. The constraint function enforces the transformation equations $T'_{ij} = \Omega'^{-1}_i T'_{ij} \Omega'_j$ to be fulfilled.

C. Extrinsic Parameter Propagation and Global Fusion

At this point, after the $N-1$ Kalman filter runs, each module M_i has $N-1$ estimations of the calibration parameters T'_{ij} within the ground plane. These $N-1$ transformations are sufficient to calculate the remaining transformations as follows:

$$\hat{T}'_{jl} = T'^{-1}_{ij} T'_{il} = T'_{ji} T'_{il} \tag{8}$$

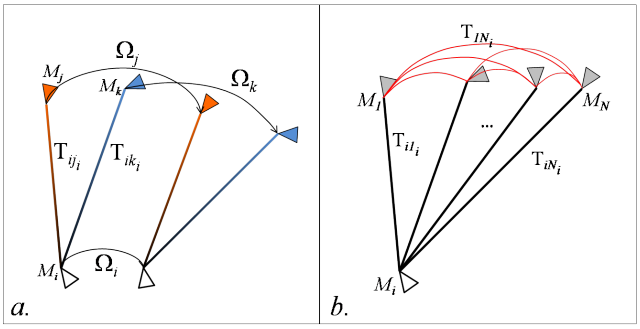

Fig. 3. *a.* Local estimations of the extrinsic parameters between module M_i and $M_{j\neq i}$ using a Kalman filter and the local sensor data. *b.* Determination of the remaining calibration parameters by state and error propagation based on the locally estimated calibration data.

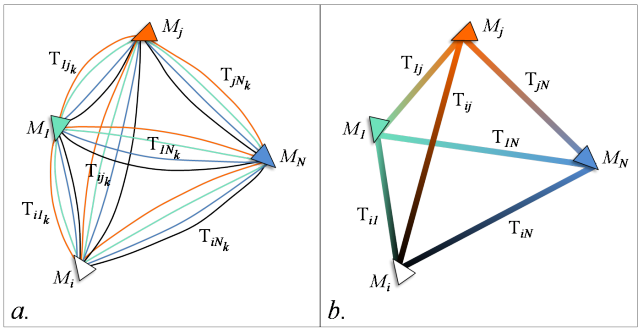

Fig. 4. *a.* After propagation, there exist $k = 1, ..., N$ local guesses for each of the $N(N-1)/2$ extrinsic transformations. *b.* The N local guesses of each transformation are fused to a global transformation using the DKF approach.

These are $(N-1)(N-2)/2$ additional calculations. As a result, each module has a local guess for each of the $N(N-1)/2$ extrinsic parameters between the cameras. This means on the other hand that there are N guesses for each extrinsic camera transformation (Fig. 4a). The covariances are propagated according to [6].

To merge these N estimates per camera pair we can again proceed just like in eq. 4 and eq. 5 (Fig. 4b).

D. Merging the Transformations

To get the 6 dof Euclidean transformation that describes the extrinsic configuration of the cameras we have to merge the transformations. The transformation that contains all 3 translational and 3 rotational parameters is the given by

$$T_{ij} = G_i^{-1}(k) \cdot T'_{ij} \cdot G_j(k). \tag{9}$$

VI. Experimental Result

We tested our implementation with simulated data. We chose a two camera rig with one front and one side looking camera. The simulated optical flow vectors are disturbed with $\sigma = 0.5$ pixels. The length of the history was chosen $H = 10$. Initially all parameters were set to zero. The motion was chosen to be of almost constant velocity and linear increasing yaw rate (Fig. 6). This special motion allows to see at which angle the estimation process starts converging.

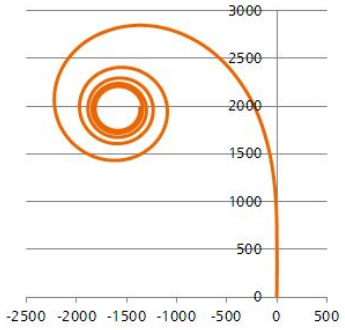

Fig. 6. Simulated motion track.

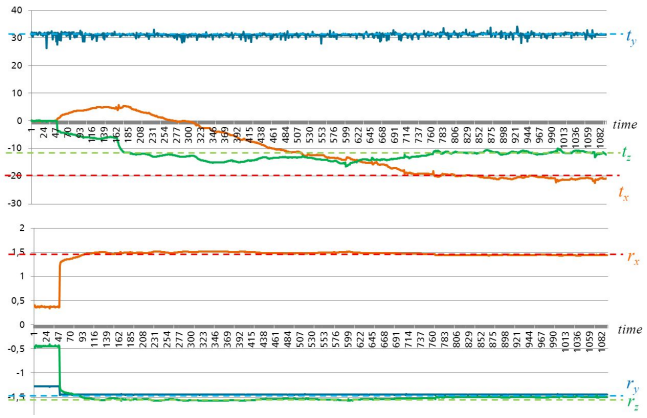

Fig. 7. Resulting extrinsic parameters (top: translation; bottom: rotation). The dotted lines indicate the correct parameter values.

Fig. 7 shows that the lateral translational parameters begin converging approx. at frame 165 and reach the final values at approx. frame 760. The rotation parameters converge quite faster, as their esrimation does not depend on the rotational motion. At frame 165 the current yaw rate was still $< 1°$. But the minimum rotation angle depends always on the size of the history H as the rotation angles are accumulated for the local parameter estimation in the ground plane. The estimation of the parameter t_y mostly depends on the ground estimation in section V-A and hence are not much influenced by the estimation process described in section V-B.

VII. Conclusion and Future Work

We presented a purely vision-based approach for calibrating multiple non-overlapping cameras in a

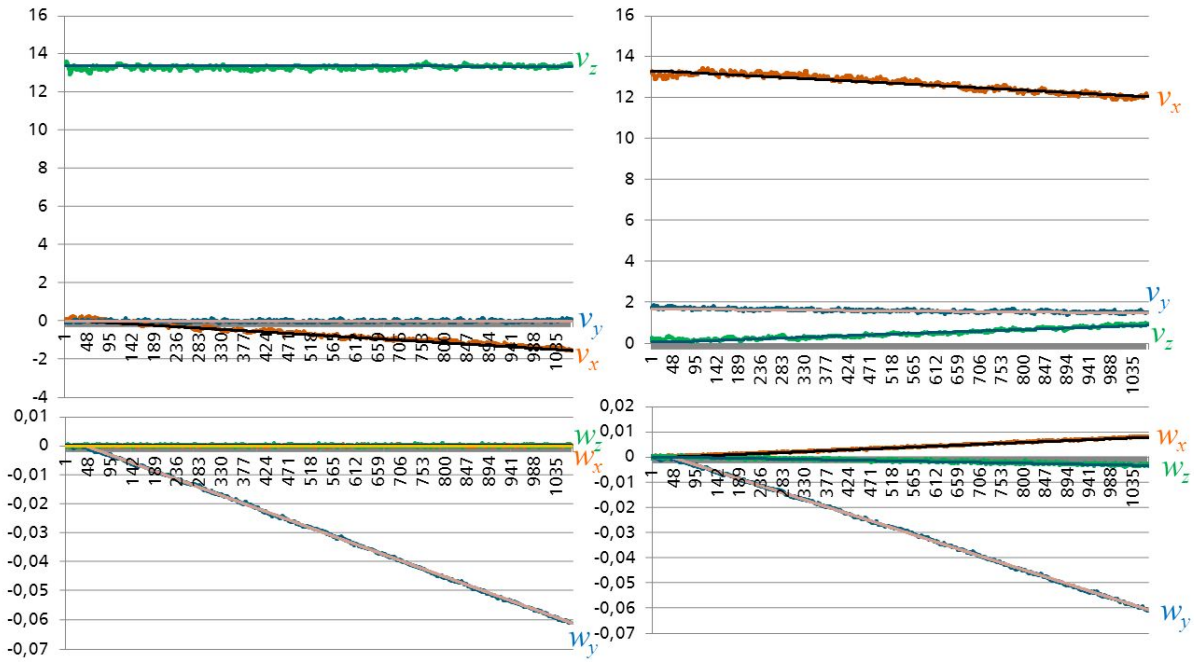

Fig. 5. Estimated motion parameters with ground truth (left: cam1; right: cam2; top: translation; bottom: rotation.

planarly moving vehicle. As the geometric structure of the whole system and hence the estimation task is quite complex, our approach accounts for the uncertainties of the calibration and motion parameters. Additionally to [6] an intermediate ground estimation step was introduced, so that even for planar motions all 6 parameters of en extrinsic transformation can be estimated. The results with a simulated test case shows that a minimum yaw rate is necessary to make the parameters converging to the real parameter values.

In the near future we are going to test the proposed approach with real data. Especially robustness and convergence capabilities are of special interest, but also the behaviour of $N > 2$ cameras. Therefore more simulated test cases will be designed and evaluated.

REFERENCES

[1] T. Dang, C. Hoffmann and C. Stiller, "Continuous stereo self-calibration by camera parameter tracking", *IEEE Transactions on Image Processing*, 2009.

[2] S. Esquivel, F. Woelk, R. Koch, "Calibration of a Multi-camera Rig from Non-overlapping Views", *DAGM-Symposium*, 2007.

[3] B. Lamprecht, S. Rass, S. Fuchs, K. Kyamakya, "Extrinsic Camera Calibration for an On-Board Two Camera System Without Overlapping Field of View", *in Proceedings of the IEEE Intelligent Transportation Systems Conference*, 2007.

[4] A. F. Mordohai, A. Akbarzadeh, J. M. Frahm, P. Mordohai, C. Engels, D. Gallup, P. Merrell, M. Phelps, S. Sinha, B. Talton, L. Wang, Q. Yang, H. Stewenius, R. Yang, G. Welch, H. Towles, D. Nister, M. Pollefeys, "Towards Urban 3D Reconstruction From Video", *in 3DPVT*, 2006.

[5] F. Pagel, "Robust Monocular Egomotion Estimation Based on an IEKF", *IEEE Canadian Conference on Computer and Robot Vision*, 2009.

[6] F. Pagel, D. Willersinn, "Motion-based Online Calibration for Non-overlapping Camera Views", *IEEE Intelligent Transportation Systems Conference*, 2010.

[7] T. Ruland, H. Loose, T. Pajdla, L. Kruger, "Hand-Eye Autocalibration of Camera Positions on Vehicles", *IEEE Intelligent Transportation Systems Conference*, 2010.

[8] R. C. Smith,P. Cheeseman, "On the Representation and Estimation of Spatial Uncertaint", *in The International Journal of Robotics Research, vol. 5 no. 4*, Dec. 1986.

[9] R. Y. Tsai, R. K. Lenz, "A New Technique for Fully Autonomous and Efficient 3D Robotics Hand/Eye Calibration", *in IEEE Transactions on Robotics and Automation, vol. 5, no. 3*, June. 1989.