2010 IEEE Intelligent Vehicles Symposium
University of California, San Diego, CA, USA
June 21-24, 2010

Calibration of Non-Overlapping Cameras in Vehicles

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

Abstract— **Cameras are getting smaller and cheaper. So a cost-effective usage of multi-camera systems in vehicles gets more and more attractive. In many cases it is desirable to cover the whole environment all around the vehicle. But often design restrictions and energy consumption do not allow constellations of cameras with overlapping field of views. However, for a common and geometric usage of the extracted information, e. g. in structure from motion tasks, it is necessary to know the relative alignment of the cameras, which are the extrinsic calibration parameters. This paper adresses the extrinsic calibration of a multi camera rig with non-overlapping field of views on a mobile platform. As the field of views do not necessarily overlap, common calibration methods based on corresponding image points between the camera views will fail. This problem can be overcome by using the mobiltiy of the platform. A pattern-based method for extrinsic calibration of the camera rig on a mobile plattform is presented.**

I. Introduction

Over last years multi camera applications in vehicles became more and more popular as the costs of the sensor production decreased drastically. Multiple sensors can be used to cover a wider field fo view, as in [2], to the point of full omnidirectional vision (fig. 1). However, in many cases it is not possible to cover the full environment or to guarantee overlapping field of views. This is because restrictions in design, energy consumption and physical capacity have to be considered.

To be able to use the full range of possibilities that go along with such a sensor constellation, the relative adjustment of the sensors must be known. The extrinsic parameters between each camera can be described as an Euclidean transformation. Then, the informations that are extracted from the single cameras can be merged and referenced in a common coordinate system. This might be very useful for reconstruction, object detection or attention guidance tasks.

This paper adresses the problem of determining the extrinsic calibration parameters between the cameras. The proposed solution uses a flexible adjustment of a set of well-known patterns to perform an offline calibration.

A. Related Work

The topic of calibrating camera networks without overlapping field of views has mostly been adressed in the context of rigid multi-sensor networks.

Kumar et al. [5] present a two-step approach to calibrate a multi-camera set extrinsically. First, they calibrate each camera intrinsically. Then, the extrinsic parameters are estimated using a common calibration pattern. They overcome the need for all cameras to see the calibration object by allowing them to see it through a mirror. They use the fact that the mirrored views generate a family of mirrored camera poses that uniquely describe the real camera pose.

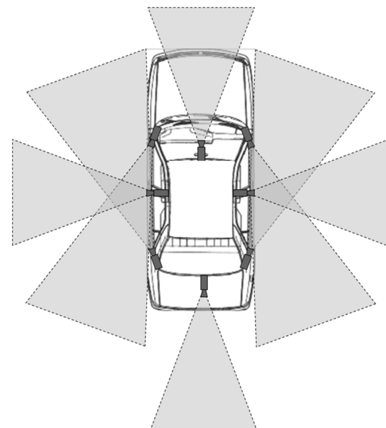

Fig. 1. Example of a vehicle with partially and nonoverlapping field of views.

Rahimi et al. [8] simultaneously localize and track (SLAT) a person walking through a rigid multi-sensor network. By assuming a motion model for the target, which means prior knowledge about the person's dynamics, they close the gap between the field of views of the cameras. They are calibrating the network of cameras by aligning each cameras ground-plane coordinate system with a global ground-plane coordinate system.

Other approaches use the mobility and the motion of the cameras explicitely. Esquivel et al. [3] defined a constraint purely based on the egomotion of each single camera. The extrinsic parameters are determined by simply using the trajectories of the cameras. Hence, the quality of the calibration result depends strictly on the quality of the motion estimation algorithm. It turns out that the calibration needs rotational motion to be able to calculate the translational components of the extrinsic parameters. So, this calibration suffers due to the almost purely planar motion of the camera-carrying vehicle. A planar motion does not allow the determination of longitudinal distances between the cameras. A practical calibration approach must overcome this drawback. The most common technique for a robust calibration is the usage of

978-1-4244-7868-2/10/$26.00 ©2010 IEEE

well-known calibration objects.

Lamprecht et al. [6] calibrate a non-overlapping two-camera rig on a vehicle. The calibration runs online. They use the motion of the vehicle and tracked patterns. Traffic signs serve as calibration objects. Their solution shows strong sensitivity concerning the accuracy of the localization of the host-vehicle.

B. Proposed Solution

Basically, the goal adressed in this paper is to find a solution to calibrate a mobile rig quick and with little effort. Furthermore, we want to be able to easily run a recalibration, even when the vehicle is somewhere outside the test area.

In principle, if a well-known calibration pattern is used that is simply big enough to cover all cameras (just like a big, hollow cylinder), the rig could be calibrated by projecting the known 3D points onto the images and minimizing the projection error. But in practice, it is difficult to arrange such an environment because some kind of "calibration garage" under laboratory conditions is needed. To be able to perform a calibration on-site, the suggestion in this paper is to use several handy objects that are placed in the scene. The patterns and their arrangement is not fully arbitrary. They must be uniquely identifiable and they should be placed so that at least two neighboring patterns are visible and two cameras see a pattern each from some point of view. This point will be further discussed in section 2. The mobility of the rig is used to determine the relative alignment of the patterns as well as the extrinsic camera parameters. The calibration is performed by moving the rig in front of the patterns (fig. 2). Because the structure of the patterns is well-known and hence their 3D positions can be determined reliably, the calibration is not restricted by any motion. Once this alignment is known, we can consider all the single patterns as a common big one and use this composite pattern to determine the extrinsic camera parameters. The geometric constraints are based on the projection error of the object points and hence very simple and straight forward.

This kind of calibration has some benefits (especially from the viewpoint of a developer). Although a sufficiently big place is needed to perform this kind of calibration (the rig/vehicle must move in front of the pattern arrangement), it might be easier to find a free area rather than to drive to the next calibration garage or back to the laboratory and loose valuable time for data acquisition and test runs. By performing a stepwise calibration, a global optimization over all measured data to determine all pattern/camera transformations in one single calculation step can be avoided.

Furthermore, a concept is presented, how the acquired image data can be evaluated and organized during the acquisition process. This might be necessary because the driver might like to know when to stop the calibration while he is driving in front of the patterns. This is the case when enough data has been acquired to calibrate all the patterns' positions as well as the cameras extrinsics among each other.

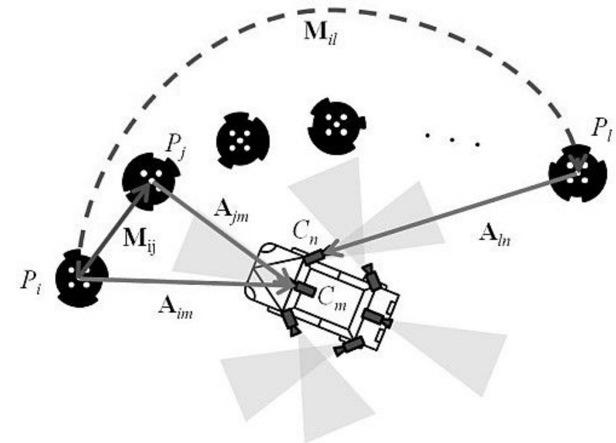

Fig. 2. Calibration scheme. After $\mathbf{M}$ is known, the extrinsics $\mathbf{C}$ of the cameras C_m, C_n can be calibrated. The parameters $\mathbf{A}$ can be calculated from the projections of the well known patterns P_i, P_j.

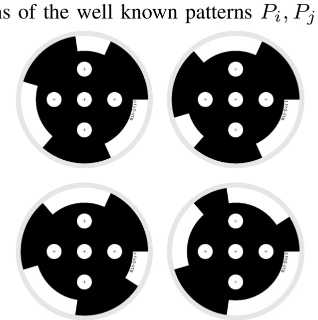

Fig. 3. MCMXT calibration patterns.

II. A Three-Step Calibration Procedure

The whole calibration routine is structured as follows: First, the intrinsic camera parameters are calculated using a standard pattern-based calibration routine, as in [9]. This process is called $Cal0$. Second, the Euclidean parameters $\mathbf{M}_{ij}$ that describe the transformations between the patterns is determined, where

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

is a homogeneous 4×4 transformation matrix with rotation $\mathbf{R}$ and translation $\mathbf{t}$. This extrinsic pattern calibration is called $Cal1$. Finally, when the intrinsic camera parameters and the relative alignement of the patterns are known, the extrinic camera paremeters $\mathbf{C}_{mn}$ can be estimated ($Cal2$), with

$$\mathbf{C}_{mn} = \begin{pmatrix} \mathbf{R}_{mn} & \mathbf{t}_{mn} \\ \mathbf{0}^T & 1 \end{pmatrix}_{4\times 4}. \tag{2}$$

A. The Pattern

For the following approach the pattern must fulfill three criteria: It must be uniquely identifiable, it must be detectable in real-time (at least 1Hz for a reasonable usage in practice) and the 3D position and orientation $\mathbf{A}_{im} \in \mathbb{R}^{4\times 4}$ of the pattern P_i relative to the camera C_m (see fig. 4 and 5)

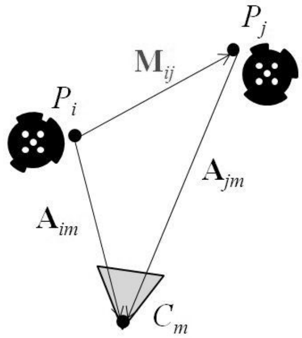

Fig. 4. Relative position of two pattern visible in one camera ($Cal1$).

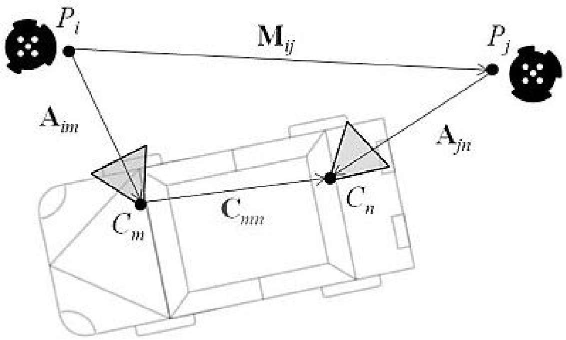

Fig. 5. Extrinsic camera parameters with known pattern alignement $\mathbf{M}_{ij}$ ($Cal2$).

must be reliably determinable. Last criteria is easily achieved by using a planar pattern with at least four points, as in [9]. To fulfill the first two criteria we use the so called MCMXT-patterns [4]. This pattern was designed to be reliably detectable in real-time, e. g. on crash test dummies. The chamfers in the outer circle are used to identify the patterns uniquely. Currently there are about 128 different IDs available.

B. Determination of the Pattern Configuration

To determine the Euclidean parameters $\mathbf{M}_{ij}$ the squared projection error is minimized. This is the distance between the measured image points and the projections of the 3D points in the respective coordinate systems of the patterns. At this point it is important that the pattern pair (P_i, P_j) is visible in one camera C_m at one time. Two neighboring patterns have to be close enough to each other so that they are visible in a single camera view. Hence, a chainwise pattern configuration is suggested.

The determination of the patterns' parameters is executed in a pairwise manner. The homogenous point $\tilde{\mathbf{v}}$ of a vector $\mathbf{v} = (v_1, ..., v_d)^T \in \mathbb{R}^d$ is given by $\tilde{\mathbf{v}} = (v_1, ..., v_d, 1)^T \in \mathbb{R}^{d+1}$. The projection of the pth point of pattern P_i in the coordinate system of camera C_m, $\mathbf{Y}_{m_{i_p}} \in \mathbb{R}^3$, is defined as a function $\pi : \mathbb{R}^4 \to \mathbb{R}^2$ via the intrinsic camera matrix

$$\mathbf{K}_m = \begin{pmatrix} f_{x_m} & 0 & c_{x_m} \\ 0 & f_{y_m} & c_{y_m} \\ 0 & 0 & 1 \end{pmatrix} : \tag{3}$$

$$\pi_m(\tilde{\mathbf{Y}}_{m_{i_p}}) := \mathbf{x}_{m_{i_p}}, \tag{4}$$

where $\mathbf{K}_m \mathbf{Y}_{m_{i_p}} =: (X', Y', Z')^T$ and $\mathbf{x}_{m_{i_p}} = (\frac{X'}{Z'}, \frac{Y'}{Z'})^T$.

With π and the known Euclidean parameters $\mathbf{A}^k_{im}, \mathbf{A}^k_{jm}$, that describe the transformation between camera and pattern, and the unknown transformation matrix $\mathbf{M}_{ij}$, we can define an error function f_{Pat}. $\mathbf{M}_{ij}$ can hence be determined using a LM-optimization [7] to minimize the constraint

$$\begin{aligned} f_{Pat}(\mathbf{M}_{ij}) = & \\ & \sum_p^M \sum_k^\tau \left(\mathbf{x}^k_{m_{j_p}} - \pi_m(\mathbf{A}^k_{im} \mathbf{M}^{-1}_{ij} \tilde{\mathbf{X}}_{j_p}) \right)^2 + \\ & \sum_p^M \sum_k^\tau \left(\mathbf{x}^k_{m_{i_p}} - \pi_m(\mathbf{A}^k_{jm} \mathbf{M}_{ij} \tilde{\mathbf{X}}_{i_p}) \right)^2 \end{aligned} \tag{5}$$

for all τ time respective measurement steps and M points per pattern. Notice that $\mathbf{M}_{ij}$ is constant for all $k \in \{1, ..., \tau\}$. Here, the square of a vector is the dot product. To initialize $\mathbf{M}_{ij}$ for the LM-optimization one can set

$$\mathbf{M}^{init}_{ij} := \mathbf{A}^{k^{-1}}_{jm} \mathbf{A}^k_{im} \tag{6}$$

for any $k \in \{1, ..., \tau\}$. Optionally, $\mathbf{M}^{init}_{ij}$ can be chosen to be the average of $\mathbf{A}^{k^{-1}}_{jm} \mathbf{A}^k_{im}$ for all k.

C. Extrinsic Camera Calibration

Once the pattern alignement is known, we can calibrate the cameras quite similar to the way we did before with the patterns. When a pattern P_i is visible in camera C_m and pattern P_j is visible in camera C_n at time k, we can determine the extrinsic alignment between C_m and C_n (fig. 5). The intrinsic matrices $\mathbf{K}_m$ and $\mathbf{K}_n$, the pattern alignment $\mathbf{M}_{ij}$ as well as the transformations between the camera and the pattern coordinate system $\mathbf{A}^k_{im}$ and $\mathbf{A}^k_{jn}$ are assumed to be known for τ measurements.

Analogously to (5) and (6), an error function with respect to $\mathbf{C}_{mn}$ can be formulated as

$$\begin{aligned} f_{Cam}(\mathbf{C}_{mn}) = & \\ & \sum_p^M \sum_k^\tau \left(\mathbf{x}^k_{n_{j_p}} - \pi_n(\mathbf{C}_{mn} \mathbf{A}^k_{im} \mathbf{M}^{-1}_{ij} \tilde{\mathbf{X}}_{j_p}) \right)^2 + \\ & \sum_p^M \sum_k^\tau \left(\mathbf{x}^k_{m_{i_p}} - \pi_m(\mathbf{C}^{-1}_{mn} \mathbf{A}^k_{jn} \mathbf{M}_{ij} \tilde{\mathbf{X}}_{i_p}) \right)^2 \end{aligned} \tag{7}$$

with an initialization

$$\mathbf{C}^{init}_{mn} := \mathbf{A}^k_{jm} \mathbf{M}_{ij} \mathbf{A}^{k^{-1}}_{im} \tag{8}$$

D. Determining the Transformation between Pattern and Camera

Zhang [9] proposes a monocular calibration method that determines the intrinsic camera parameters as well as the extrinsic parameters that describe the transformation between pattern and camera. This section will shortly sum up the basic principle of determining the extrinsic transformation $\mathbf{A}_{im}$.

Given a rotation matrix $\mathbf{Q} = (\mathbf{q}_1\mathbf{q}_2\mathbf{q}_3)$, a translation vector $\mathbf{c}$ and an intrinsic camera matrix $\mathbf{K}$, the projection of a homogeneous 3D point $\tilde{\mathbf{X}}_p = (X, Y, Z, 1)^T$ is given up to scale by

$$\lambda \mathbf{x}_p = \lambda \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \mathbf{K}[\,\mathbf{q}_1\,\mathbf{q}_2\,\mathbf{q}_3\,\mathbf{c}\,]\tilde{\mathbf{X}}_p. \tag{9}$$

The calibration contributed in this paper deals with planar calibration patters. So all p points are assumed to lie on a plane. Then we can reformulate (9) as follows:

$$\lambda \mathbf{x}_p = \mathbf{K}[\,\mathbf{q}_1\,\mathbf{q}_2\,\mathbf{q}_3\,\mathbf{c}\,] \begin{pmatrix} X \\ Y \\ 0 \\ 1 \end{pmatrix}_p = \mathbf{K}[\,\mathbf{q}_1\,\mathbf{q}_2\,\mathbf{c}\,] \begin{pmatrix} X \\ Y \\ 1 \end{pmatrix}_p . \tag{10}$$

So we can define the homography

$$\mathbf{H} = \mathbf{K}[\,\mathbf{q}_1\,\mathbf{q}_2\,\mathbf{c}\,] = [\,\mathbf{h}_1\,\mathbf{h}_2\,\mathbf{h}_3\,]. \tag{11}$$

$\mathbf{H}$ can be estimated with at least four point correspondences between the pattern and the image plane (see e. g. [1]). As the intrinsic parameters are assumed to be known, the extrinsic transformation is given by

$$\begin{aligned} \mathbf{q}_1 &= \lambda \mathbf{K}^{-1}\mathbf{h}_1 \\ \mathbf{q}_2 &= \lambda \mathbf{K}^{-1}\mathbf{h}_2 \\ \mathbf{q}_3 &= \mathbf{q}_1 \times \mathbf{q}_2 \\ \mathbf{c} &= \lambda \mathbf{K}^{-1}\mathbf{h}_3 \end{aligned} \tag{12}$$

with

$$\lambda = \frac{1}{\mathbf{K}^{-1}\mathbf{h}_1} = \frac{1}{\mathbf{K}^{-1}\mathbf{h}_2}. \tag{13}$$

The resulting rotation parameters can be used for a further refinement to numerically calculate a rotation matrix $\mathbf{Q}^\star$ that fulfills the claim $\mathbf{Q}^{\star T}\mathbf{Q}^\star = \mathbf{I}$ by minimizing the Frobenius norm

$$min_{\mathbf{Q}^\star} \left\|\mathbf{Q}^\star - \mathbf{Q}\right\|_F^2$$

(see [9] for further details).

III. Data Acquisition

This section covers the question: When do we have to stop data acquisition? More precisely, when was enough date acquired so that *Cal1* and *Cal2* can be executed. This part is quite important to avoid multiple calibration sessions. Because without an automatic data management it may happen, for example, that after a session the acquired data is not sufficient to determine the extrinsic parameters between two patterns. Hence, the whole session would have to be repeated.

There are definitely several possibilities to manage the captured data. Here, a method is proposed that simply increases entries of an adjacency matrix $\mathbf{J}$: One for the pattern ($\mathbf{J}_{Pat}$) and one for the cameras ($\mathbf{J}_{Cam}$). To acquire the data for the pattern calibration *Cal1* one can proceed as follws (the data acquisition for the extrinsic camera parameters *Cal2* is analogous):

The upper right matrix is initialized with zeros. Each time two patterns P_i, P_j are visible in view C_m, $\mathbf{J}_{Pat}\big(\min(i,j), \max(i,j)\big)$ is incremented by 1. If P_i is visible in C_m and P_j in C_n, $\mathbf{J}_{Cam}\big(\min(m,n), \max(m,n)\big)$ is incremented by 1. The data acquisition can be stopped when every column $(1, ..., N-1)$ has at least one entry $\geq K$. K is the predefined minimum number of views that must be acquired for the calibration. Thinking of a graph where the patterns/cameras are vertices, every vertex v_i is connected to a vertex v_j with $i > j$. This means that each vertex is reachable from any other vertex of the graph. So, the acquired data is enough to determine the extrinsic parameters between every pattern and every camera.

Once $\mathbf{J}_{Pat}$ is known, the computation of the transformation between the patterns that have a K-entry in $\mathbf{J}_{Pat}$ can be done directly by minimizing (5). The calculation of all the other transformations is then done iteratively:

As transformation $\mathbf{M}_{0,1}$ (that is the edge $P_0 \to P_1$) must be known (given by the structure of the adjacency matrix), the transformations are determined by proceeding from left to right in the matrices.

There are two cases to distinguish. Whether the row index of a K-entry in $\mathbf{J}_{Pat}$ in column j is $j-1$ (or in other words: K is on a "*step*" in $\mathbf{J}_{Pat}$), or it is not. In the latter case, given $\mathbf{M}_{i,j}$ with $j - i > 1$, we can determine the corresponding *step*-transformation as follows:

$$\mathbf{M}_{j-1,j} = \mathbf{M}_{i,j}\mathbf{M}_{i,j-1}^{-1} \tag{14}$$

Proceeding from left to right through $\mathbf{J}_{Pat}$ guarantees that $\mathbf{M}_{i,j-1}$ in the previous column is already known. This leads to the first case. Given $\mathbf{M}_{j-1,j}$ we can calculate $\mathbf{M}_{j-k-1,j}$ iteratively for all $k = 1, ...j-1$:

$$\mathbf{M}_{j-k-1,j} = \mathbf{M}_{j-k,j}\mathbf{M}_{j-k-1,j-k} \tag{15}$$

Furthermore we have $\mathbf{M}_{i,j} = \mathbf{M}_{j,i}^{-1}$ and $\mathbf{M}_{i,i} = \mathbf{I}_{4\times4}$. To calculate all transformations $\tilde{\mathbf{C}}_{m,n}$ it can be proceeded in the same way with $\mathbf{J}_{Cam}$.

IV. Results

A. Simulated Data

To investigate the influence of noise on the accuracy of the calibration process, test series were run with simulated data for a two-camera rig. Hence, the calibration error ϵ is defined as the square root of the mean squared projection error (RMSE) according to (5) and (7):

$$\epsilon_{Pat} = \sqrt{\frac{f_{Pat}(\mathbf{M}_{ij})}{M \cdot \tau}} \tag{16}$$

$$\epsilon_{Cam} = \sqrt{\frac{f_{Cam}(\mathbf{C}_{mn})}{M \cdot \tau}} \tag{17}$$

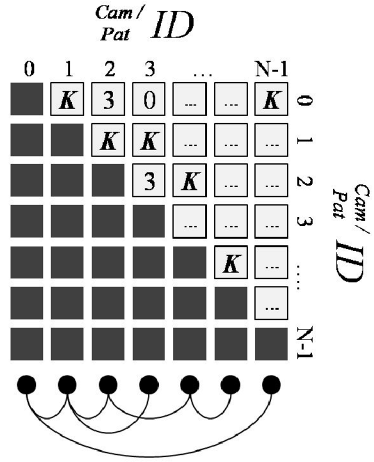

Fig. 6. Adjacency matrix $\mathbf{J}_{Pat}$ respective $\mathbf{J}_{Cam}$. The end of data acquisition is reached when each column has at least one entry $\geq K$. Then each vertex can be reached from every vertex of the graph.

Therefore different noise influences like image noise, focal length errors, rotational and translational errors were considered (fig. 9 to fig. 12). So, for each noise level, the calibration process was repeated 20 times and the results were averaged. The simulation used a pattern size of $M = 80$ and $\tau = 9$. The calibration results for the patterns $(\mathbf{M}_{ij})$ and the cameras $(\mathbf{C}_{mn})$ were evaluated independently. For the simulation nine camera positions $\mathbf{A}^k_{im}$ were defined. Image size is 640×480 and the intrinsic parameters were defined as $f_x = 580, f_y = 580$ for C_m and $f_x = 600, f_y = 600$ for C_n and the principal point $(c_x = 320, c_y = 240)$ for both cameras.

Obviously noisy transformation parameters have the greatest influence of the calibration error, namely rotation and translation ($\mathbf{A}_{im}$ and $\mathbf{M}_{ij}$ from $Cal1$ for $Cal2$). Especially errors in the rotatation parameters lead to enormous calibration errors in $Cal2$. When looking at (7) or fig. 5, the transformation path is obviously longer than in fig. 4 and hence the influence of Euclidean transformation errors is bigger. Noisy image positions of the projected pattern points or a noise focal length have only little effect on the calibration result compared to erroneous transformation parameters. So a robust calulation of the Euclidean transformations $\mathbf{A}^k_{im}$ between the patterns and the cameras is eminent for the success of this calibration method.

B. Real Data

To test the calibration method in practice, a camera rig was build up. It consisted of two intrinsically calibrated cameras mounted with an angle of $\approx 45°$ and an approximate camera baseline of $\approx 11.5cm$ (see fig. 7, both ground truth values were measured by hand). Four patterns (fig. 8) and a minimum of five data sets for a calibration routine were used ($K = 5$, for $Cal1$ as well as for $Cal2$). The camera rig was moved in front of the patterns by hand.
The resulting angle was $42.3°$ and a baseline of $12.01cm$. The final error was $\epsilon_{Cam} = 1.21$, which is in most cases a

Fig. 7. The camera rig demonstrator. The cameras are mounted with an angle of $\approx 45°$.

Fig. 8. Test scenery with four patterns.

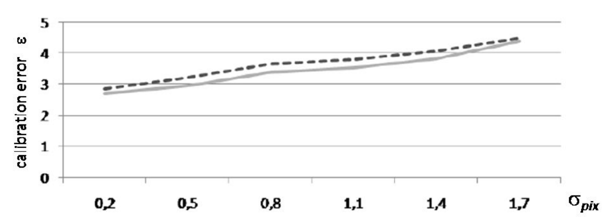

Fig. 9. Calibration error dependend to image noise σ_{pix}. The noise is set to $\sigma_{foc} = 1.0, \sigma_{rot} = 0.1$ and $\sigma_{trans} = 0.01$. The result of the pattern calibration *Cal1* is shown by the solid line, the result of the camera calibration *Cal2* is shown by the dotted line.

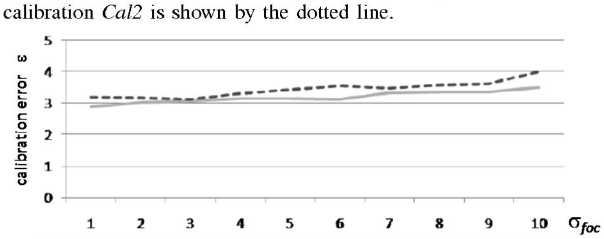

Fig. 10. Calibration error dependend to focal length errors σ_{foc}. The noise is set to $\sigma_{pix} = 0.5, \sigma_{rot} = 0.1$ and $\sigma_{trans} = 0.01$. The result of the pattern calibration *Cal1* is shown by the solid line, the result of the camera calibration *Cal2* is shown by the dotted line.

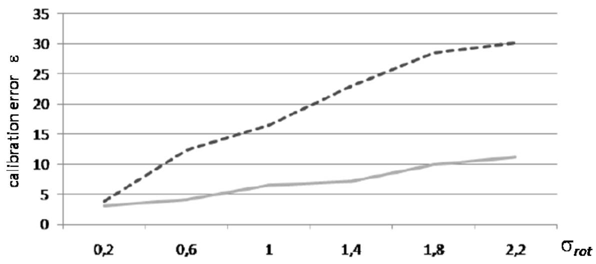

Fig. 11. Calibration error dependend to rotational errors σ_{rot}. The noise is set to $\sigma_{foc} = 1.0, \sigma_{pix} = 0.5$ and $\sigma_{trans} = 0.01$. The result of the pattern calibration *Cal1* is shown by the solid line, the result of the camera calibration *Cal2* is shown by the dotted line.

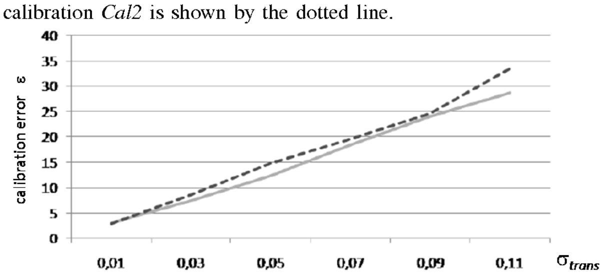

Fig. 12. Calibration error dependend to translational errors σ_{trans}. The noise is set to $\sigma_{pix} = 0.5, \sigma_{foc} = 1.0$ and $\sigma_{rot} = 0.1$. The result of the pattern calibration *Cal1* is shown by the solid line, the result of the camera calibration *Cal2* is shown by the dotted line.

toleratable remaining error for such a sensor constellation. This error should at least be sufficiently accurate, so that the parameters can serve as an initial guess for an online calibration routine.

V. Conclusions and Future Works

A. Conclusions

In this paper a pattern based calibration method was presented. The calibration procedure follows three steps. First, a standard intrinsic calibration for each camera. Second, the extrinsic pattern configuration is recovered. Third, the extrinsic camera parameters are calculated. After step one, the actual calibration procedure starts. The data acquisitions for step 2 and 3 are executed simultaneously. Therefore mobile rig moves in front of the patterns. Concurrently, the acquired data is evaluated to determine, when enough data has been acquired to estimate the Euclidean parameters.

In contrast to probabilistic approaches that use e. g. mobile patterns with a dynamic motion model, no assumptions about any motions have to be made. This is because of the usage of well-known patterns, all constraints base on simple geometric projections. Furthermore, an error-prone motion estimation can be avoided. And a planar motion does not lead to any restrictions in the degrees of freedom that can be estimated. The pattern configuration is quite simple and hence very flexible.

B. Future Works

As especially motion-based systems suffer from physical instability because of vibrations and material deformation due to temperature changes, the next logical step is an online calibration. Such a calibration makes use of the proposed offline calibration as an initial parameter guess. The Euclidean transformation parameters are filtered and optimized during runtime. So, annoying periodic recalibration procedures can be (hopefully) avoided. Currently, developments progress and results will be shown in future proceedings.

References

[1] A. Agarwal, C.V. Jawahar, P.J. Narayanan, "A Survey of Planar Homography Estimation Techniques", *Technical Reports, International Institute of Information Technology, Hyderabad (Deemed University)*, 2005.

[2] B. Clipp, J.H. Kim, J.M. Frahm, M. Pollefeys, R.I. Hartley, "Robust 6DOF Motion Estimation for Non-Overlapping, Multi-Camera Systems", *in Proceedings of the 2008 IEEE Workshop on Applications of Computer Vision*, 2008.

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

[4] German-Patent: 10 2006 060 716.3-53 "Verfahren zur Bildmarken-unterstützten Bildauswertung"

[5] R.K. Kumar, A. Ilie, J.M. Frahm, M. Pollefeys, "Simple calibration of non-overlapping cameras with a mirror", *in Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*, 2008.

[6] 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.

[7] W. Press, S. Teukolsky, W. Vetterling, B. Flannery , "Numerical Recipes in C", *2nd ed. Cambridge, UK: Cambridge University Press*, 1992.

[8] A. Rahimi, B. Dunagan, T. Darrell, "Simultaneous calibration and tracking with a network of non-overlapping sensors", *in Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*, 2004.

[9] Z. Zhang, "A flexible new technique for camera calibration", *in IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2000.