# Calibration of a Multi-Camera Rig From Non-Overlapping Views

Sandro Esquivel, Felix Woelk, and Reinhard Koch

Christian-Albrechts-University, 24118 Kiel, Germany

**Abstract.** A simple, stable and generic approach for estimation of relative positions and orientations of multiple rigidly coupled cameras is presented in this paper. The algorithm does not impose constraints on the field of view of the cameras and works even in the extreme case when the sequences from the different cameras are totally disjoint (i.e. when no part of the scene is captured by more than one camera). The influence of the rig motion on the existence of a unique solution is investigated and degenerate rig motions are identified. Each camera captures an individual sequence which is afterwards processed by a structure and motion (SAM) algorithm resulting in positions and orientations for each camera. The unknown relative transformations between the rigidly coupled cameras are estimated utilizing the rigidity constraint of the rig.

## 1 Introduction

Rigidly coupled cameras with non overlapping views appear in many scenarios: In the automotive industry f.e. rear view cameras and blind spot cameras gain popularity, sewer inspection systems equipped with two antipodal cameras are commercial available and also in surveillance applications multiple non-overlapping cameras are used. In many of these situations the relative position of these cameras is of interest.

General methods estimating these rig parameters assume that the cameras have overlapping views such that points lying in these views can be used to register the positions of the cameras with each other [1, 2]. This paper suggests an approach for rig parameters estimation from non-overlapping views using sequences of time-synchronous poses of each camera. Such poses can be obtained from SAM algorithms on synchronously captured image sequences. The presented approach works in three stages:

**Internal camera calibration:** First, the internal calibration of each camera on the rig is computed using standard techniques [4]. The internal camera calibration consists of the focal length, principal point, skew and lens distortion parameters.

**Pose estimation:** Second, the external pose of each camera in the rig is computed for each frame in arbitrary coordinate systems using SAM techniques [5, 1]. Note that without further knowledge the geometry can only be reconstructed up to scale and hence the coordinate systems of the reconstructions of the cameras are related by a similarity transform.

**Rig calibration:** The scale of each coordinate system and the internal positions and orientations of the rigidly coupled cameras are estimated using constraints between poses resulting from the previous stage. Nonlinear optimization techniques can be used for refinement.

The paper is organized as follows: After reviewing previous work, the theoretic foundation of the algorithm is explained. Degenerate cases are identified and solutions for these cases are suggested. Finally experiments with synthetic and real data are presented.

## 2 Previous Work

Sequence reconstruction algorithms profit from rigidly coupled cameras. Frahm e.a. proposed a method for stabilizing 3D scene reconstruction by utilizing images of a moving rig [3]. Broader views of the scene could be reconstructed by using a multi-camera system. In order to perform such a task one has to determine the relative transformations between the cameras of the rig in addition to the intrinsic parameters of each camera (i.e. focal length and principal point). There are many approaches registering the poses of a set of cameras with each other. Most of these approaches, such as metric calibration of a stereo-rig [2], rely on an overlapping field of view. An approach to align non-overlapping image sequences has been made by Caspi and Irani [6] for the case of multi-camera systems sharing the same projection center, but not for general multi camera system. The task at hand is closely related to the field of hand-eye-estimation such as [7] which faces a similar problem: The (fixed) relation between poses measured in different coordinate frames, e.g. between a sensor mounted onto a robot's hand and the hand itself must be estimated. In a similar manner a multi-camera system with cameras fixed in a rig can be interpreted as a hand-eye system where a motion sensor is lacking but pose information can be retrieved from multiple image sequences.

## 3 Theoretical Background

In the following, superscripts are used to identify a specific time and subscripts are used to identify a specific camera in the rig. For example $C_i^\kappa$ denotes the center of projection of camera $i$ at time $\kappa$.

### 3.1 Rigid Transformations

The change between two Cartesian reference frames is described by a *similarity transformation*. Each similarity transformation is of the form

$$T = \begin{pmatrix} \lambda R & C \\ 0^T & 1 \end{pmatrix} \tag{1}$$

where $\lambda \in \mathbb{R}$ accounts for the different scales of coordinate systems, $R \in \mathbb{R}^{3\times3}$ is an orthogonal rotation matrix describing the relative orientation and $C \in \mathbb{R}^3$ is the translation between the two reference frames. When the scale $\lambda$ is equal to 1, $T$ is also called *Euclidean transformation.* Using projective space, the change of reference frame of a projective point vector can be achieved by simple matrix vector multiplication $X^{target} = TX^{source}$. The concatenation of two subsequent changes of reference frames $T_1$ and $T_2$ can be computed by a simple matrix multiplication

$$T = T_2 T_1 \tag{2}$$

### 3.2 Reference Frames and Transformations

The reconstructions from each individual camera are usually given in separate reference frames. The different reference frames are defined next.

**Camera Reference Frames:** We assume that each reconstruction is described in the coordinate system whose origin and orientation matches the position and orientation of the first camera and whose scale is given such that the baseline between the first two cameras equals 1 as our SAM algorithm delivers. The pose of each camera $i$ at each time $\kappa$ is described in the reference frame of the reconstruction by its orientation $R_i^\kappa$ and position $C_i^\kappa$. Obviously the initial pose of each camera is then given by $R_i^0 = I$ and $C_i^i = (0\ 0\ 0)^T$. These reference frames are denoted as *camera* coordinate system. Each physical camera in the rig has an associated camera reference frame.

**Local Reference Frames:** Obviously the choice of the first camera for the definition of the camera reference frame is somewhat arbitrary. Any other time $\kappa \neq 0$ could be chosen for the definition of position and orientation of reference frame resulting in the *local* coordinate system. The Euclidean transformation $T_i^\kappa$ relating the camera reference frame with the $i$-th local reference frame is given by

$$T_i^\kappa = \begin{pmatrix} R_i^\kappa & C_i^\kappa \\ 0^T & 1 \end{pmatrix} \tag{3}$$

A local reference frame can be defined for each frame from each sequence resulting in an overall of $m = KN$ reference frames. Here $N$ denotes the number of cameras and $K$ denotes the number of frames in each sequence.

**Global Reference Frame:** Working with multiple reference frames easily becomes confusing and error-prone. Hence without the loss of generality a dedicated *master camera* is chosen and the associated reference frame is chosen as the *global* coordinate system. The master camera is identified with the subscript index $i = 0$. All other cameras are denoted as *slave* cameras.

### 3.3 Relations Between Reference Frames

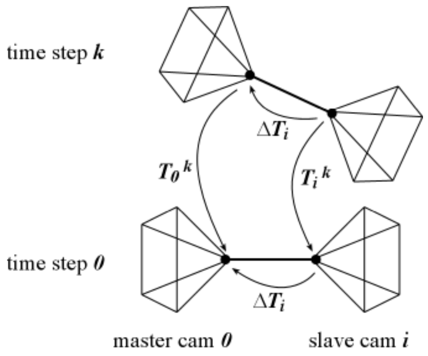

**Fig. 1.** Relations between cameras in the rig

The transformation between the global reference frame and each local reference frame of the slave camera $i$ at time $\kappa$ can be computed in two alternate ways. Either by first transforming to the local reference frame of the master camera at time $\kappa$ and afterward using the unknown similarity transforming $\Delta T_i$ to get to the local reference frame of the slave camera $i$ at time $\kappa$

$$T_0^\kappa \Delta T_i \tag{4}$$

or alternatively by first changing into the camera reference frame $\Delta T_i$ and afterward using the Euclidean transform $T_i^\kappa$ to get to the destination

$$\Delta T_i T_i^\kappa \tag{5}$$

This relation is illustrated in fig. 1.

## 4 Estimation of the Rig Parameters From Poses

Eq. 4 and 5 must result in the same transformation and hence

$$T_0^\kappa \Delta T_i = \Delta T_i T_i^\kappa \tag{6}$$

must hold for each time $\kappa = 1, \ldots, K$ and each slave camera $i = 1, \ldots, N$. Equation 6 can be decomposed into one constraint regarding only orientations

$$R_0^\kappa \Delta R_i = \Delta R_i R_i^\kappa \tag{7}$$

and one constraint linking both orientations and positions

$$R_0^\kappa \Delta C_i + C_0^\kappa = \Delta\lambda_i \Delta R_i C_i^\kappa + \Delta C_i \tag{8}$$

Note that the scale $\Delta\lambda_i$ has no influence on eq. 7 because it appears on both sides. Except from the scale factor, this result equals the well-known relation between the different coordinate frames in the hand-eye calibration problem [7], where the master camera defines the sensor frame and the slave camera defines the hand frame.

### 4.1 General Motion

When the motion of the rig is general, i.e. when it rotates and translates, a two step approach is feasible. First the orientation is recovered using eq. 7 and afterward it is utilized for the recovery of position and scale using equation 8.

**Recovery of Orientation:** There has been extensive work on solving orientation equations such as Eq. 7. Early solutions [8] represent rotations by 3 × 3-rotation matrices resp. 9-vectors resulting in straight forward linear formulations. These approaches tend to be error-prone and suffer from difficulties in enforcing the orthogonality constraint on the resulting matrix. Seminal contribution such as [9] represent rotations by unit quaternions and hence reduce the number of variables from 9 to 4. Further on, the unit length constraint for quaternions is far simper to enforce than orthogonality [10]. Replacing the rotation matrices by quaternions $q$ in Eq. 7 results in

$$q_0^\kappa \cdot \Delta q_i = \Delta q_i \cdot q_i^\kappa \quad \text{or equivalently} \quad (T_{q_0^\kappa} - T^*_{q_i^\kappa})\Delta q_i = 0 \tag{9}$$

where $T_q, T_q^*$ define left and right multiplication with quaternion $q = (w, x, y, z)^T$ [11]

$$T_q = \begin{pmatrix} w & -x & -y & -z \\ x & w & -z & y \\ y & z & w & -x \\ z & -y & x & w \end{pmatrix} \qquad T_q^* = \begin{pmatrix} w & -x & -y & -z \\ x & w & z & -y \\ y & -z & w & x \\ z & y & -x & w \end{pmatrix} \tag{10}$$

Hence we derive the following linear equation system for the unknowns $\Delta q_i = (\Delta w_i, \Delta x_i, \Delta y_i, \Delta z_i)^T$ at each time step $\kappa = 1, \ldots, K$ subject to $|\Delta q_i| = 1$

$$\underbrace{\begin{pmatrix} w_0^\kappa - w_i^\kappa & -x_0^\kappa + x_i^\kappa & -y_0^\kappa + y_i^\kappa & -z_0^\kappa + z_i^\kappa \\ x_0^\kappa - x_i^\kappa & w_0^\kappa - w_i^\kappa & -z_0^\kappa - z_i^\kappa & y_0^\kappa + y_i^\kappa \\ y_0^\kappa - y_i^\kappa & z_0^\kappa + z_i^\kappa & w_0^\kappa - w_i^\kappa & -x_0^\kappa - x_i^\kappa \\ z_0^\kappa - z_i^\kappa & -y_0^\kappa - y_i^\kappa & x_0^\kappa + x_i^\kappa & w_0^\kappa - w_i^\kappa \end{pmatrix}}_{A_i^\kappa} \begin{pmatrix} \Delta w_i \\ \Delta x_i \\ \Delta y_i \\ \Delta z_i \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \tag{11}$$

Apparently one pair of corresponding poses for each camera suffices for the estimation of the internal rotation parameters when the rig motion includes sufficient orientation change. When the rig is purely translating, eq. 11 degenerates and can no longer be solved. A detailed investigation of the degenerate case of purely translating motion is presented in section 4.2. Computation of the rotation matrices from unit quaternions can be found in [11]. The quaternion constraint can be explicitely modelled by using the Lagrangian multiplier as described in [7].

**Recovery of Position and Scale:** Once an estimate for the internal rotation $\Delta R_i$ of slave camera $i$ has been found, the internal position $\Delta C_i$ and scale $\Delta\lambda_i$ can be found by solving the linear constraint of eq. 8 resulting in the linear system

$$\underbrace{\begin{pmatrix} I - R_0^\kappa & \Delta R_i C_i^\kappa \end{pmatrix}}_{B_i^\kappa} \begin{pmatrix} \Delta C_i \\ \Delta\lambda_i \end{pmatrix} = C_o^\kappa \tag{12}$$

Equation system 12 consists of 3 equations per pose correspondence and 4 unknowns and hence at least 2 corresponding pose pairs for each slave camera are necessary for a unique solution.

### 4.2 Pure Translation

When the rig motion is purely translational, the relative orientations $q_0^\kappa$ and $q_i^\kappa$ in eq. 11 are both given by the quaternion representing zero rotation $(1,0,0,0)^T$ and the matrix $A_i^\kappa$ becomes zero. Even when the rotation of the rig is very small, $A_i^\kappa$ is close to zero and the equation system 11 becomes ill-conditioned[1]. Fortunately this situation can easily be detected simply by looking at the orientations $R_i^\kappa$. The estimation of the $C_i^\kappa$ is not possible in this case, however internal orientation and scale can still be estimated. Assuming that the local rotations $R_i^\kappa$ are each equal to $I$ and considering only the directions $\boldsymbol{c}_i^\kappa = \frac{C_i^\kappa}{\|C_i^\kappa\|}$, eq. 8 becomes

$$\varDelta R_i \boldsymbol{c}_i^\kappa = \boldsymbol{c}_0^\kappa \tag{13}$$

which can be solved linearly in closed form using the quaternion representation [10]. Because a rotation does not change the length of a vector, the scale can be estimated without knowledge about $\varDelta R_i$. Mean and variance of the scale are computed using poses from different times $\kappa$.

$$\varDelta\lambda_i^\kappa = \frac{|C_0^\kappa|}{|\varDelta R_i C_i^\kappa|} = \frac{|C_0^\kappa|}{|C_i^\kappa|} \qquad \varDelta\lambda_i = \sum_n^N \frac{\varDelta\lambda_i^n}{N}, \quad \sigma_{\lambda_i}^2 = \sum_n^N \frac{(\varDelta\lambda_i^n - \varDelta\lambda_i)^2}{N} \tag{14}$$

### 4.3 Nonlinear Refinement

It is obvious that errors in the estimation of the internal rotation will inflict the estimation of the internal translation and scale. Once an estimate for the rig parameters has been found via the LLS approach as described in section 4.1, nonlinear refinement can be used to simultaneously estimate internal orientation, position and scale. The error functional

$$f(\varDelta q_i, \varDelta C_i, \varDelta\lambda_i) = \sum_{\kappa=1}^K |A_i^\kappa \varDelta q_i|^2 + |B_i^\kappa \begin{pmatrix} \varDelta C_i \\ \varDelta\lambda_i \end{pmatrix} - C_0^\kappa|^2 \tag{15}$$

is minimized using a Levenberg-Marquardt.

### 4.4 MAP Refinement

Experiments on real image data revealed that the error functional (15) is very sensitive to noise. However the situation improved when the scale was held fixed at an approximate value during estimation. To circumvent this problem, the error functional 15 is augmented by a maximum a posteriori (MAP) term for the scale resulting in

$$f(\varDelta q_i, \varDelta C_i, \varDelta\lambda_i) = \sum_{\kappa=1}^K |A_i^\kappa \varDelta q_i|^2 + |B_i^\kappa \begin{pmatrix} \varDelta C_i \\ \varDelta\lambda_i \end{pmatrix} - C_0^\kappa|^2 + \frac{(\varDelta\lambda_i - \lambda_i)^2}{\sigma_{\lambda_i}^2} \tag{16}$$

with the prior guess of the scale $\lambda_i$ and uncertainty $\sigma_{\lambda_i}^2$ computed using eq. 14.

[1] This is also visible in eq. 12 where $B_i^\kappa$ grows ill-conditioned when the rotation $R_0^\kappa$ observed by the master camera is close to $I$

## 5 Experiments

Experiments on synthetic pose data precede experiments on synthetic image data and finally experiments on real image sequences are presented.

### 5.1 Synthetic Pose Data

Synthetic pose data corrupted with normal distributed error is used for the tests. It is generated as follows: First $N$ random rig parameters $\Delta R_i$, $\Delta C_i$, $\Delta \lambda_i$ and $K$ random master poses $R_0^\kappa$, $C_0^\kappa$ are generated. Afterwards the associated slave poses are computed by applying the rig parameter to the master pose and rotating the resulting ground truth slave pose by $\varepsilon$ degrees around a randomly chosen axis to simulate errors resulting from the pose estimation process.

**Linear Model Comparison:** To compare both linear algorithms (i.e. the purely translational model and the general motion model) the errors are computed under a variety of different conditions, namely different orientations and different input error accuracies. Figure 2 compares the errors of both models and illustrates the equal error boundary for both algorithms.

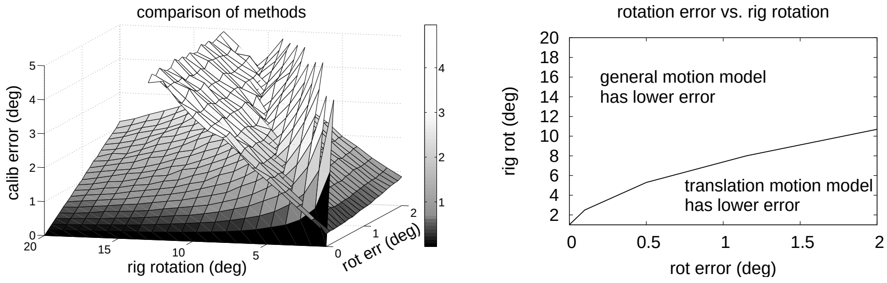

**Fig. 2.** Comparison of the two linear models. (a): Errors of both models on dependency of rotation and input orientation accuracy. (b): Equal error boundary for general motion model and rotation only model. In conditions above the equal error boundary, the general motion model yields more accurate results than the purely translational model.

**Sensitivity to Input Pose Errors:** To analyze the dependency of the estimation results on noise of the input data, tests on a large number of randomly generated input poses are performed. Figure 3 shows the average resulting orientation error of the estimated internal orientations and translations dependent on the input error $\varepsilon$ for the linear general motion approach and for the nonlinear optimization. Four input pose pairs were used for each test. The calibration error grows approximately linearly with the input pose error $\varepsilon$.

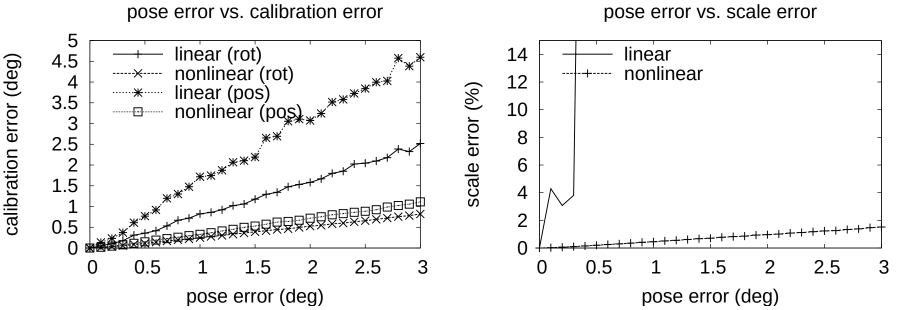

**Fig. 3.** Sensivity of the algorithm to errors in the input poses. The resulting rotation and translation orientation error (a), and scale error (b) are shown vs. input pose orientation error for linear solution and the nonlinear refinement.

### 5.2 Synthetic Image Sequences

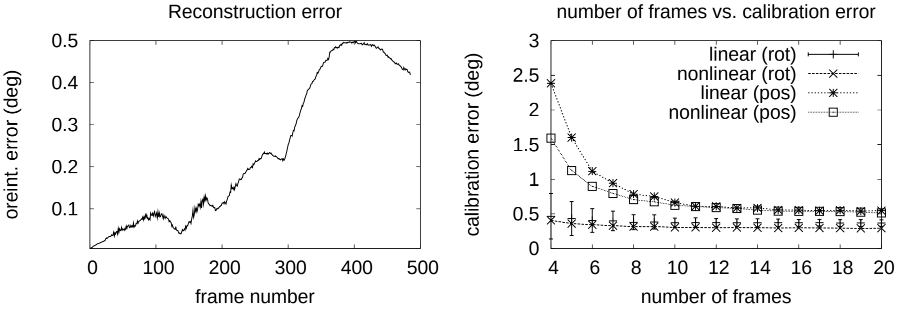

**Fig. 4.** (a) Orientation error of SAM algorithm on synthetic image sequence. See text for details. (b) Calibration error (orientation and translation) vs. number of randomly selected input poses on synthetically rendered images. The average error over 1000 tries is plotted.

A synthetic rig consisting of two cameras moves in a synthetic scene resulting in two sequences of synthetically generated images consisting of 500 frames each. The SAM results are transformed in a global coordinate system such that the rig constraints strictly hold on the first two frames. Note that the pose estimates from the SAM algorithm have a rotation error of up to 0.4 degree (figure **??**(a)). $\Delta R_i^\kappa = R_i^\kappa * (R_0^\kappa)^T$ is estimated from the two orientations for each frame and the error with the respect to ground truth is plotted for each frame in fig. **??**(a).

The dependency of the calibration error on the number of input poses (i.e. frames) is shown in figure **??**(b) for the linear estimation methods with general motion model and for the nonlinear estimation. The input poses again derive from the SAM results on the synthetic image sequences. It can be seen that the estimation results do not improve significantly for $K \geq 10$.

### 5.3 Real Image Sequences

Two physical setups were investigated: One with overlapping views for the comparison with a marker based algorithm, and a non-overlapping sequence for demonstration purposes.

**Overlapping Sequence:** The physical setup consists of two cameras at a distance of approximately 30cm with a relative yaw angle of about 60°. The rig calibration is computed using (i) the calibration toolbox from [12] resulting in external poses for each frame. The relative transform is computed robustly as the average over 24 frames. Additionally the rig calibration is estimated (ii) using the suggested MAP refinement algorithm. To enhance stability a RANSAC algorithm is used in combination with our approach. The orientation difference between the two results (i) and (ii) was 0.62°, the direction difference of the two resulting translation vectors was 1.52°, and the translation length error was about 1.33 percent (see table 1(a)). The rig calibration result is in the same order of magnitude as the result from the marker based approach.

**Table 1.** Rig calibration results for (a) overlapping and (b) non-overlapping sequence.

| method | orientation | position | scale |
|---|---|---|---|
| (a) Bouget | $56.94° \pm 0.54°$ | $(25.2 \pm 0.2, -3.9 \pm 0.3, 11.5 \pm 0.06)$ | $28\text{cm} \pm 0.66\text{cm}$ |
| (a) our approach | $57.56°$ | $(25.8, -3.2, 11.4)$ | 28.37cm |
| (b) our approach | $158.14°$ | $(0.9, -7.4, -13.3)$ | 15.29cm |

**Non-overlapping Sequence** For the non-overlapping sequence the cameras were rotated approx. about 160° with respect to each other and set up at with a distance of about 15cm. Figure 5 shows a photo of the rig and a 3D model with the reconstructed rig parameters. The rig internal translation is roughly along the optical axis such that the cameras look in opposite directions. The rig was rotated around its center and translated slightly parallel to the image planes such that the sequences do not overlap. The estimated internal rig rotation was 158.14° around axis $(0.4, 0.74, 0.51)^T$, and the internal translation direction was estimated to be $(0.06, -0.48, -0.87)^T$ with length 15.29cm (see table 1(b)). The resulting calibration meets the expectation by qualitative evaluation. Future work will include tests on non-overlapping sequences with ground truth data available.

## 6 Conclusions

A novel approach for the estimation of rig parameters using non-overlapping sequences was introduced. Two nonlinear refinement algorithms for the rig parameters have been proposed and tested on synthetic poses and on synthetic and

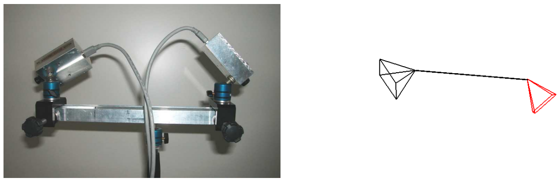

**Fig. 5.** Photo of real rig (a) and 3D model of the rig calibration estimate (b).

real image sequences. It has been shown that the achievable accuracy resides in the same order of magnitude as marker based approaches achieve. In addition, the calibration can also be achieved in a non-overlapping setup where no marker calibration is possible. Of course the accuracy is dependent on the results of the SAM algorithm.

**Future Work:** Future work could investigate the benefit of direct integration of the calibration process into the SAM algorithm. Also other methods circumventing the dependency of the calibration on SAM results should be found and investigated. Because we do not depend on visual image information, poses received from sensor data can also be utilized for camera alignment.

## References

1. R. Hartley, A. Zisserman: Multiple View Geometry in Computer Vision. Cambridge University Press, Cambridge (2000)
2. A. Zisserman et. al.: Metric Calibration of a Stereo Rig. Proc. WRVS (1995)
3. J.M. Frahm et. al.: Pose Estimation for Multi-Camera Systems" Proc. DAGM (2004)
4. R. Y. Tsai: A Versatile Camera Calibration Technique for High-Accuracy 3D Machine Vision Metrology Using Off-the-Shelf TV Cameras and Lenses. IEEE Jour. RA **3(4)** (1987) 323–344
5. M. Pollefeys et. al.: Visual Modeling with a Hand-Held Camera, IJCV **59(3)** (2004) 207–232
6. Y. Caspi, M. Irani: Alignment of Non-Overlapping Sequences. Proc. ICCV, (2001)
7. R. P. Horaud, F. Dornaika: Hand-Eye Calibration. IJRR **14(3)** (1995) 195–210
8. Y. C. Shiu, S. Ahmad: Calibration of Wrist Mounted Robotic Sensors by Solving Homogeneous Transform Equations of the Form $AX = XB$. IEEE Jour. RA **5(1)** (1989) 16–29
9. J. C. K. Chou and M. Kamel: Finding the Position and Orientation of a Sensor in a Robot Manipulator Using Quaternions. IJRR **10(3)** (1991) 240–254
10. B. K. P. Horn: Closed-Form Solution of Absolute Orientation Using Unit Quaternions. J. Opt. Soc. Am. A **4(4)** (1987) 629 ff.
11. W. Foerstner and B. Wrobel: Mathematical Concepts in Photogrammetry. In: J. C. McGlone (ed.): Manual of Photogrammetry, 5th edition. ASPRS (2004) 47–49
12. J.Y. Bouguet: Camera Calibration Toolbox for Matlab `http://www.vision.caltech.edu/bouguetj/calib_doc/index.html`