A Simulation Study and Experimental Verification of Hand-Eye-Calibration using Monocular X-Ray

Petra Dorn[1], Peter Fischer[1,2], Holger Mönnich[2], Philip Mewes[2], Muhammad Asim Khalil[3], Abhinav Gulhar[2], Andreas Maier[1]

[1]Lehrstuhl für Mustererkennung, FAU Erlangen-Nürnberg
[2]Siemens Healthcare GmbH, Forchheim, Germany
[3]K-tronik GmbH, Garching
petra.dorn@fau.de

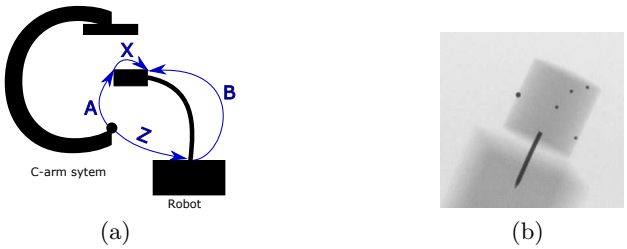
Abstract. In this paper, the simultaneous hand-eye/robot-world problem $\boldsymbol{AX} = \boldsymbol{ZB}$ is performed using a single X-ray image instead of a stereo camera in order to avoid the additional tracking device. Our setup consists of a special X-ray marker, several image preprocessing steps, and a monocular pose estimation algorithm, for extracting the 6-D pose of the marker with respect to the X-ray source. Simulations are performed to investigate the behavior of the proposed hand-eye method when including inaccuracies of the robot and the non-isotropic errors of monocular pose estimation. The simulations were evaluated in an experimental setup, reaching an accuracy of 0.06° and 0.77 mm.

1 Introduction

In surgical interventions robots are on the advance because of their high precision and repeatability. A common topic in robotics is known as hand-eye calibration and covers the procedure of finding the rigid transformation between the camera or marker mounted onto the robot's end-effector and the robot flange, denoted as $\boldsymbol{X}$. The state-of-the-art procedure to solve this problem is performed with optical tracking. However, additional equipment is needed. In this paper, the hand-eye calibration is performed using a C-arm system which is typically available in many operating rooms that require imaging. The theory behind the calibration can be solved with many mathematical approaches, which can differ, for example, in their parametrization for the rotation and their choice of solving rotation and translation either simultaneously or separately. Zhuang et al. and Dornaika et al. solve the problem $\boldsymbol{AX} = \boldsymbol{ZB}$ with the help of quaternions and compute the orientation before translation [1,2]. Shah rewrites the rotational part in terms of a Kronecker product [3]. Li et al. present two methods where they solve rotation and translation simultaneously. Besides the Kronecker product they use dual quaternions for the second method [4]. Iterative methods, which often use the

The concepts and information presented in this paper are based on research and are not commercially available.

Fig. 1. a) Model for the problem $\boldsymbol{AX} = \boldsymbol{ZB}$, and b) 2-D acquisition of the marker.



(a) (b)

Levenberg-Marquardt algorithm for finding the minimum of the optimization problem, are presented e.g. in [5]. Each paper is based on the classical setup where either a camera is mounted on the end-effector of the robot and acquires a calibration pattern or an optical tracking system acquires a marker attached on the robot. In this paper, the tracker is replaced by a static C-arm system which acquires 2-D images of the marker. Problems arising from monocular instead of stereo view are the inaccuracies in estimation of the depth and out-of-plane rotations [6]. In simulations, the effect of the expected inaccuracies from the robot and from pose estimation of the marker with respect to the X-ray source is investigated. These results are compared with real measurements.

2 Methods

The simultaneous hand-eye and robot-world problem $\boldsymbol{AX} = \boldsymbol{ZB}$ is visualized in Figure 1(a). The problem describes a closed loop with the two unknown matrices $\boldsymbol{X}$ and $\boldsymbol{Z}$ which is the transformation between marker and robot flange or X-ray source and robot base, respectively. During the calibration procedure the robot flange moves to different poses, resulting in several measurements $\boldsymbol{A}_i$ – transformations from the C-arm source to the marker – and $\boldsymbol{B}_i$ – transformations from the robot base to the flange. While $\boldsymbol{B}$ is known due to the forward kinematic of the robot, $\boldsymbol{A}$ has to be estimated from the X-ray image. Each matrix of the equation $\boldsymbol{A}_i\boldsymbol{X} = \boldsymbol{Z}\boldsymbol{B}_i$ is a rigid transformation matrix which includes a 3×3 rotation matrix $\boldsymbol{R}$ and a 3×1 translation vector $\boldsymbol{t}$. $\boldsymbol{AX} = \boldsymbol{ZB}$ can be split into a rotational and a translational equation

$$\boldsymbol{R_A R_X} = \boldsymbol{R_Z R_B} \tag{1}$$

$$\boldsymbol{R_A t_X} + \boldsymbol{t_A} = \boldsymbol{R_Z t_B} + \boldsymbol{t_Z} \ . \tag{2}$$

For finding the pose of the marker $\boldsymbol{A}$, the 3-D geometric model of the marker is required. Our marker has a cylindrical shape and consists of five metal beads embedded in plastic. Metal has a much higher density than plastic and can be seen easily on X-ray images (Figure 1(b)). In order to estimate the 6-D pose from a 2-D image, which is also known as Perspective-n-point (PnP) problem, at least four metal beads have to be visible on the 2-D acquisition. The fifth bead of the

marker is redundant, but might be necessary when two dots overlap each other. The pose estimation algorithm provided by the Visual Servoing Platform (ViSP), a modular cross platform library written in C++, is used [7]. The included dot tracker algorithm finds dots on an image automatically, and checks their shape and size. The further pose estimation algorithm, which is based on the ideas of DeMenthon and Lowe, matches a 3-D model with the projection, finds the best combination, and calculates the transformation matrix [8,9]. Since there might be some disturbances on the X-ray image, which impair the success of the algorithm, different preprocessing steps are necessary. First, a black top-hat filter is used in order to remove large objects in the image. Second, an intensity normalization is performed. Third, this filtered image is binarized with an adaptive threshold. These implementations make use of the OpenCV library. All steps are done in order to highlight the dots and remove other disturbing objects in the image. The challenge of pose estimation is that the dimensions which mean a change in depth of the beads are more difficult to estimate. These are the z-coordinate which describes the point on the axis between X-ray source and detector, α and β which are the rotations around the x- and y-axis of the X-ray source frame.

3 Experiments

The hand-eye calibration was performed with Octave using the open source implementation from Shah [3]. The experiments are conducted both on simulated and real data.

3.1 Hand-Eye Simulation

The assumed geometrical distances in the synthetic setup are based on a realistic arrangement of robot, marker, and C-arm system. The movements of the robot flange were set to a range of $\pm 50\,\text{mm}$ and $\pm 50°$ that was randomly sampled. After the ideal loops $\boldsymbol{A}_i^* \boldsymbol{X}^* = \boldsymbol{Z}^* \boldsymbol{B}_i^*$ with 15 measurement poses are created, Gaussian noise was added on the robot and C-arm data. The inaccuracies of the robot were simulated with a standard deviation $\sigma_{trans} = 0.2\,\text{mm}$ for each component of the translation and $\sigma_{rot} = 0.05°$ for each Euler angle α, β and γ. Since no valid information regarding the inaccuracies of the transformations between C-arm source and marker are given, the noise of them was varied. In the first experiment σ_{trans} and σ_{rot} were slightly increased from 0 to 1 mm and 0 to 0.5°, respectively. This was done in order to generally examine the effects of increasing noise on the accuracy of hand-eye calibration. The second experiment takes the expected inaccuracies of the monocular pose estimation into account. That means that α, β and z are assumed to be worse compared to the other components. It was of interest, whether only the overall noise or the increasing noise on single components is of importance as well. The noise level of the C-arm data was set constant to overall standard deviations $\sigma_{trans,all} = 1\,\text{mm}$ and $\sigma_{rot,all} = 0.5°$. With the factor f, α, β and z are weighted with values from 0 to

10. Thus, the standard deviations of the C-arm data were set to

$$\sigma_{trans} = \frac{1}{\sqrt{f^2+2}} \cdot (1,1,f)\, mm \tag{3}$$

$$\sigma_{rot} = \frac{\sqrt{0.5}}{\sqrt{2f^2+1}} \cdot (f,f,1)^\circ \ . \tag{4}$$

In case of synthetic data the ground truth is known. Thus, the estimated transformation matrix $\widehat{\boldsymbol{X}}$ can be compared directly with the correct matrix $\boldsymbol{X}^*$. The rotational error is computed as

$$\Delta\theta = \theta(\boldsymbol{R}_{\boldsymbol{X}^*}^{-1} \boldsymbol{R}_{\widehat{\boldsymbol{X}}}) \ , \tag{5}$$

where θ is the rotation angle given in degrees from its angle-axis representation. The translational error $\|\Delta \boldsymbol{t}\|$ in mm is the norm of the difference translation vector of ground truth and noisy matrix

$$\|\Delta \boldsymbol{t}\| = \left\| \boldsymbol{t}_{\boldsymbol{X}^*} - \boldsymbol{t}_{\widehat{\boldsymbol{X}}} \right\| \ . \tag{6}$$

3.2 Verification Measurements

For real measurements an industrial robot (KR 10 R1100 sixx, KUKA) and a C-arm system (ARTIS pheno, Siemens Healthineers) were used. The images have a spatial resolution of 0.16 mm per pixel. The results were compared with the simulated data, in order to figure out the inaccuracies of $\boldsymbol{A}$ and the resulting deviations of $\boldsymbol{X}$ and $\boldsymbol{Z}$. The errors of real data can be determined as the offset

$$\boldsymbol{E}_i = (\boldsymbol{Z}\boldsymbol{B}_i)^{-1}(\boldsymbol{A}_i\boldsymbol{X}) \ , \tag{7}$$

when following each i-th closed loop of n test data. From this error matrix $\boldsymbol{E}_i$ the rotational error in degrees is computed with

$$\mathrm{RMSE}_{rot} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\theta(\boldsymbol{R}_{\boldsymbol{E}i}))^2} \ , \tag{8}$$

while the translational error in mm is

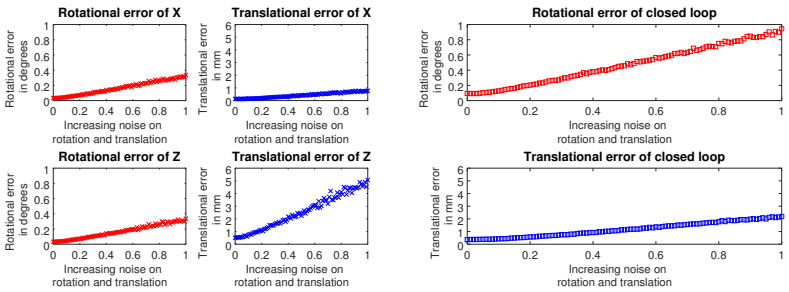
$$\mathrm{RMSE}_{trans} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\boldsymbol{t}_{\boldsymbol{E}i}^T\boldsymbol{t}_{\boldsymbol{E}i}} \ . \tag{9}$$

4 Results

4.1 Hand-Eye Simulation

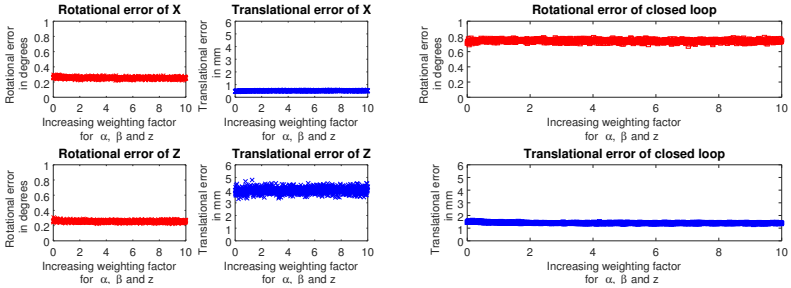
With increasing rotational and translational noise on the X-ray data, the errors both of each matrix (Figure 2(a)) and the closed loop errors (Figure 2(b))

Fig. 2. Behavior of the rotational and translational errors with increasing amount of noise on $\boldsymbol{A}$, which is the transform between C-arm and marker.



(a) Errors for each matrix (b) Errors of closed loop

Fig. 3. Behavior of the rotational and translational errors with increasing weighting factor on α, β and z of $\boldsymbol{A}$, which is the transform between C-arm and marker.



(a) Errors for each matrix (b) Errors of closed loop

increase steadily. Although the rotational error of $\boldsymbol{X}$ and $\boldsymbol{Z}$ is equal, the translational error of $\boldsymbol{Z}$ is higher. Due to the larger distance between C-arm and robot of $\|\boldsymbol{t_Z}\| = 1007.8\,\text{mm}$, compared to the distance between robot flange and marker of $\|\boldsymbol{t_X}\| = 86.6\,\text{mm}$, the same rotation error leads to higher deviations from the correct position. In the second experiment, the noise level was constant while the distribution between the single coordinates and angles was changed. The results in Figure 3 show an almost constant error level, what indicates that the overall noise level is much more important than the distribution on the components.

4.2 Verification Measurements

The 24 measurements were evaluated with 6-fold cross validation, whereby five subsets were used as training data for the hand-eye calibration and one subset as test data for calculating the errors. The averaged errors of the closed loop are: $\text{RMSE}_{rot} = 0.06°$ and $\text{RMSE}_{trans} = 0.77\,\text{mm}$. In order to reach the same results with synthetic data, the overall noise level of the C-arm data had to be set approximately to $\sigma_{trans,all} = 0.45\,\text{mm}$ and $\sigma_{rot,all} = 0.001°$. Applying these

standard deviations on simulated data and computing the errors of each matrix results in the following values: The rotational error of $\boldsymbol{X}$ and $\boldsymbol{Z}$ is $\Delta\theta = 0.04°$, the translational error of $\boldsymbol{X}$ is $\|\Delta\boldsymbol{t}\| = 0.24\,\text{mm}$, and the one of $\boldsymbol{Z}$ is $\|\Delta\boldsymbol{t}\| = 0.63\,\text{mm}$. The translational error of $\boldsymbol{Z}$ is still small because the rotational error is very low.

5 Discussion and Conclusion

In this paper, it could be shown that hand-eye calibration with X-ray is feasible. Moreover, the accuracy of the setup used in this paper seems to be similar to the optical tracking system [10]. The translational errors depend on a large extend on the overall rotational noise and the geometrical distance. To reach good results the estimated rotation has to be quite accurate. The inaccuracies of the depth estimation are tolerable as long as the other components can be computed accurately enough. In future work, the knowledge of the more inaccurate components could be used in order to further improve the pose estimation algorithm, e.g., by residual weighting.

References

1. Zhuang H, Roth ZS, Sudhakar R. Simultaneous robot/world and tool/flange calibration by solving homogeneous transformation equations of the form AX=YB. IEEE Trans Rob Autom. 1994;10(4):549–554.
2. Dornaika F, Horaud R. Simultaneous robot-world and hand-eye calibration. IEEE Trans Rob Autom. 1998;14(4):617–622.
3. Shah M. Solving the robot-world/hand-eye calibration problem using the Kronecker product. J Mech Robot. 2013;5(3):031007–1–031007–7.
4. Li A, Wang L, Wu D. Simultaneous robot-world and hand-eye calibration using dual-quaternions and Kronecker product. Int J Phys Sci. 2010;5(10):1530–1536.
5. Hirsh RL, DeSouza GN, Kak AC. An iterative approach to the hand-eye and base-world calibration problem. In: IEEE Int Conf Robot Autom. vol. 3. IEEE; 2001. p. 2171–2176.
6. Maier A, Choi JH, Keil A, et al. Analysis of vertical and horizontal circular C-arm trajectories. In: Proc. SPIE Vol. 7961; 2011. p. 7961231–7961238.
7. Marchand É, Spindler F, Chaumette F. ViSP for visual servoing: a generic software platform with a wide class of robot control skills. IEEE Robot Autom Mag. 2005;12(4):40–52.
8. Dementhon DF, Davis LS. Model-based object pose in 25 lines of code. Int J Comput Vis. 1995;15(1):123–141.
9. Lowe DG. Robust model-based motion tracking through the integration of search and estimation. Int J Comput Vis. 1992;8(2):113–122.
10. Ernst F, Richter L, Matthäus L, et al. Non-orthogonal tool/flange and robot/world calibration. Int J Med Robot. 2012;8(4):407–420.