3D-EasyCalib™ - Toolkit for the Geometric Calibration of Cameras and Robots

Darko Vehar[1,2], Rico Nestler[1,3], Karl-Heinz Franke[1]

[1]Zentrum für Bild- und Signalverarbeitung e. V.,
Werner-von-Siemens-Straße 10, 98693 Ilmenau
[2]TU Ilmenau, Fakultät für Informatik und Automatisierung,
[3]TU Ilmenau, Fakultät für Maschinenbau, FG QBV,
Postfach 10 05 65, 98684 Ilmenau
eMail: darko.vehar@zbs-ilmenau.de, rico.nestler@tu-ilmenau.de,
karl-heinz.franke@zbs-ilmenau.de

Abstract. The use of multi-modal and multi-camera systems for the optical acquisition of 3D data is widespread in the field of robotics and human-robot cooperation (HRC). In such scenarios, the 2D and 3D sensor data must be registered and merged geometrically and cameras as well as scene objects have to be precisely localised in the environment. Methods for geometric calibration of cameras and of objects acting in the environment, such as robots, play an important role for such tasks. The purpose of the geometric calibration is the estimation of the (intrinsic) imaging properties of cameras as well as the geometric position and orientation of cameras or robots in a common coordinate system using extrinsic parameters.

In this article, selected applications of geometric calibration, such as calibration of single cameras, camera pairs, of inverse cameras (projectors) and of camera-robot systems are discussed theoretically and evaluated using a modular toolkit, the 3D-EasyCalib™.

1 Introduction and motivation

Multi-camera systems are used for a variety of tasks in robotics and HRC. In such applications, before the actual 3D data analysis, a 2D, 3D fusion of all data captured by sensors in a common coordinate system and a camera-to-robot registration are required. The proven methods for camera and camera-to-robot calibration use geometrically known calibration targets that are usually recorded from different viewing directions. From known world to image point correspondences, camera parameters as well as the spatial relationships of sensors and scene objects, such as robots, can be determined very precisely. Since diverse methods often have to be combined to calibrate a specific layout, e.g. a camera-robot system, the calibration process is time-consuming and the evaluation of the calibration results is difficult. The modular toolkit 3D-EasyCalib™ was developed at the ZBS e.V. for the user-friendly, universal implementation of several tasks of geometric calibration in various scenarios.

The 3D-EasyCalib™ software offers numerous toolboxes, methods and a user-friendly front end. The aim of the toolkit is to carry out geometrical calibrations with

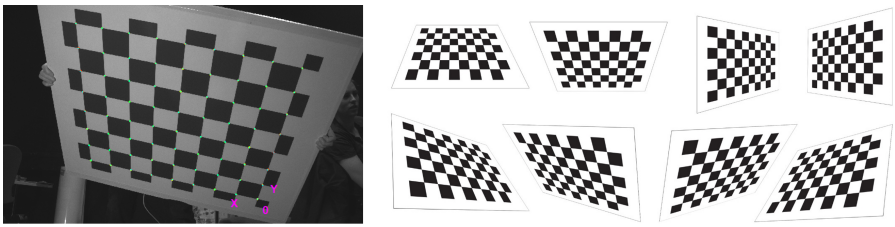

Fig. 1: Planar calibration target with 9 $\times$ 8 object points (left) for camera calibration. Eight recommended target poses for a calibration data set (right).

high precision without expert knowledge and to simplify the selection of methods suitable for the specific application. For this purpose, unified error metrics were implemented and wide-spread planar calibration targets (see Fig. 1) are used.

2 Calibration target and evaluation of target images

The prerequisite for good calibration is a precisely manufactured calibration target. A classic target provides several spatially ordered points with a known geometric position in space (3D target) or on a plane (2D or planar target). The design of the control points and their surroundings on the target have to be chosen so that they can be detected and measured as robustly, clearly and automatically as possible based on their geometry and contrasts in the camera image. From a practical point of view, targets that can be produced in different sizes without difficulty and that can be used universally for many calibration tasks are preferred. A planar calibration target with a checkerboard pattern, as shown in Fig. 1 on the left, fulfills these requirements. The calibration points are located at the corners of the squares. Unlike circular marks, these corners are invariant to lens distortion [1] and therefore can also be used to estimate and correct lens distortion.

In the 3D-EasyCalib™ the calibration target is recognized on the basis of features of the searched geometrical shape [2]. A classic, gradient-based approach [3] or the Radon method [4] is available for calculating the control points with subpixel accuracy. The second method is particularly suitable for noisy images.

To ensure high quality calibration data, the planar calibration target should be captured with different positions and with different orientations in 3D space. Since practical targets are generally smaller than the measuring space, they should be captured at different distance to camera so that the entire camera viewing frustum is covered. Varying the orientation around all three spatial axes of the target eliminates degenerate cases in which the image plane is parallel to the target. An example of eight recommended target poses, which are then captured at different distances from the camera, is shown in Fig. 1, right.

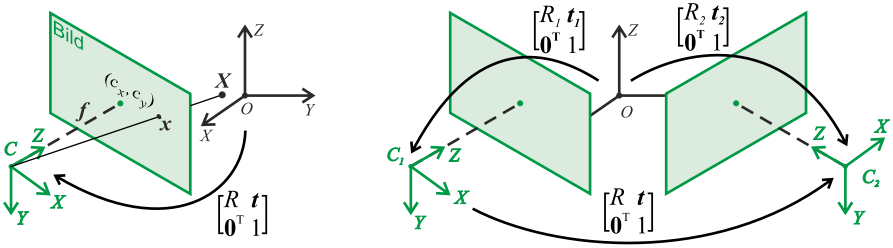

Fig. 2: Pinhole camera model (left), extrinsic parameters between two cameras (right).

3 Pinhole camera model and intrinsic calibration

3.1 Geometric definition of the pinhole camera

The pinhole camera model defines the transformation of the three-dimensional Euclidean space onto a two-dimensional image plane according to the principle of central projection and is used to approximate real cameras. In a pinhole camera, a point in space is projected onto an image point by means of a ray that goes through the projection center of the camera. If first the 3D coordinate system of the world is set equal to the 3D coordinate system of the camera with the origin at point C, then when using a notation according to [5] this projection is described compactly with the equation

$$\boldsymbol{x} = K[I|\mathbf{0}]\boldsymbol{X_C}\,. \tag{1}$$

$\boldsymbol{X_C}$ and $\boldsymbol{x}$ denote homogeneous coordinate vectors of the point in space and the resulting image point, respectively. These are defined up to a scale factor. The index C(amera) shows that the 3D point $\boldsymbol{X_C}$ is defined in the camera coordinate system. The 3×3 matrix K

$$K = \begin{bmatrix} f & \gamma & c_x \\ 0 & f & c_y \\ 0 & 0 & 1 \end{bmatrix} \tag{2}$$

is referred as the camera matrix or the calibration matrix of the pinhole camera model. It consists of the intrinsic parameters: effective focal length f, the distance of the image plane from the camera center, the principal point (c_x, c_y) and the shear parameter γ. If the point $\boldsymbol{X}$ is defined in a 3D world and the orientation and the position of the camera in this world are generally described with the rotation matrix R and the translation $\boldsymbol{t}$, then $\boldsymbol{X}$ must first be transformed affine into the camera coordinate system $\boldsymbol{X_C} = \left[\begin{smallmatrix} R & \boldsymbol{t} \\ \mathbf{0}^T & 1 \end{smallmatrix}\right]\boldsymbol{X}$ and is then projected onto the image plane according to the equation (1) as shown in the figure 2 on the left. Taking these extrinsic parameters into account, the well-known algebraic relationship of the pinhole camera model is derived

$$\boldsymbol{x} = K[R|\boldsymbol{t}]\boldsymbol{X}\,. \tag{3}$$

This model can be used not only for visual cameras, but for all imaging devices that generate the image according to the principle of central projection or projectors.

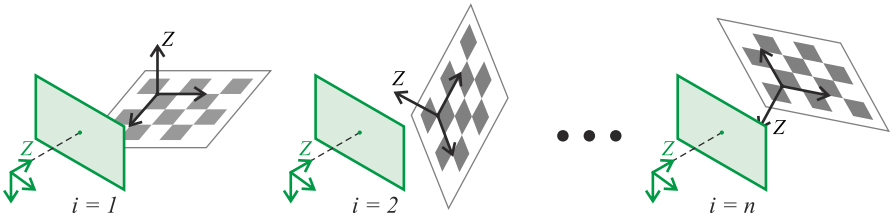

Fig. 3: The calibration data set contains several images of the target in different poses.

Exceptions are the so-called affine cameras, cameras with projection centers at infinity, which e.g. occur if telecentric lenses are used. This special case of perspective mapping must be treated separately.

3.2 Intrinsic camera calibration

In the case of an intrinsic calibration, the camera matrix K (2) is determined from images of a calibration target, its known world points $\boldsymbol{X_i}$ and the corresponding image points $\boldsymbol{x_i}$, by solving equation (3). In the community, the method of Zhang [6] has established itself particularly because of the freely available implementations, such as the easy-to-use Matlab implementation by Bouguet [7] and its port to OpenCV [2]. According to this approach, the first step is to estimate the target poses with the homography between the target and image plane. Then n recordings of the calibration target (see Fig. 3) and m known coordinates of the calibration points $\boldsymbol{X_j}$ on the target and their mapping $\boldsymbol{x_{ij}}$ on the image, are used to minimize the functional

$$\sum_{i=1}^{n}\sum_{j=1}^{m}\|\boldsymbol{x_{ij}} - \boldsymbol{x}(K, R_i, \boldsymbol{t_i}, \boldsymbol{X_j}\|^2 \,. \tag{4}$$

The Euclidean distances contained in the (4) between the calibration points projected according to the model $\boldsymbol{x}(K, R_i, \boldsymbol{t_i}, \boldsymbol{X_j})$ and the points measured in the image $\boldsymbol{x_{ij}}$ are called *reprojection errors*. This error metric, specified for each calibration point and as a total error (RMS) for each calibration target pose, is an indication of the quality of a calibration and is shown accordingly in the toolkit, see Fig. 4, left. Since the coordinates of the control points can be detected imprecisely or incorrectly, e.g. due to occlusion, shadows, camera noise or movement during the recording, potential outliers are marked in color. On this basis, these can be removed manually from the calibration data set.

If lens distortion occurs, the pinhole camera model can be expanded with parameters to describe the lens distortion and then optimized non-linearly according to the same principle (4).

4 Extrinsic calibration of multiple cameras (Stereo)

The aim of the extrinsic calibration is to determine the relative position $\boldsymbol{t}$ and orientation R of cameras to each other, e.g. with a stereo configuration of two cameras. The

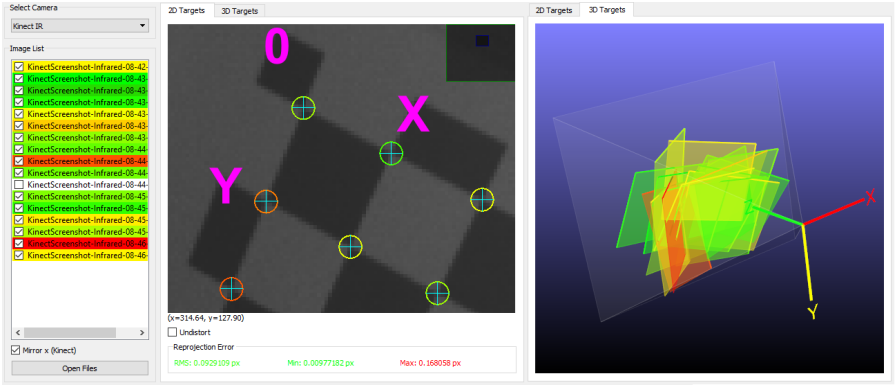

Fig. 4: Intrinsic calibration of a KinectV2 IR camera. On the left side of the window is the list of calibration images. These are shown in color according to the size of the calibration error, from green (min error) to red (max error). In the middle is the image of a selected target pose with detected calibration points (circles) and the points computed according to the camera model (crosses). The color of the circles encodes the absolute difference between these points. The various target poses in the camera's coordinate system are displayed on the right.

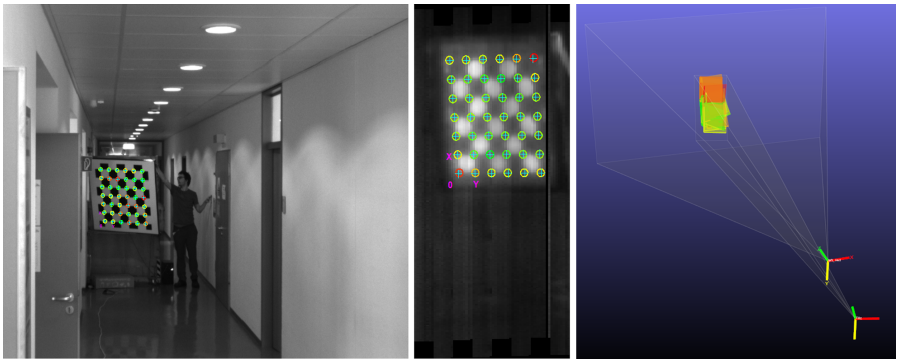

Fig. 5: Extrinsic calibration of an imaging THz scanner (body scanner) and a monochrome camera. On the left is the image of the target taken with a standard camera and in the middle the image of the same target recorded with the THz camera. The 3D view on the right uses the camera coordinate systems to show the relative orientation and position of these two sensors to another as well as the different target poses.

target is recorded simultaneously by several cameras, the target pose is calculated from the perspective of the cameras and the Euclidean transformation we are looking for, see Fig. 2, on the right, is derived as

$$\begin{bmatrix} R & \boldsymbol{t} \\ \boldsymbol{0^T} & 1 \end{bmatrix} = \begin{bmatrix} R_2 & \boldsymbol{t_2} \\ \boldsymbol{0^T} & 1 \end{bmatrix} \begin{bmatrix} R_1 & \boldsymbol{t_1} \\ \boldsymbol{0^T} & 1 \end{bmatrix} . \tag{5}$$

With the resulting extrinsic parameters $R = R_2 R_1^T$ and $\boldsymbol{t} = \boldsymbol{t_2} - R\boldsymbol{t_1}$, 3D points can be transformed from the coordinate system of the first camera into the coordinate system of the second camera and projected virtually onto the image plane.

The nonlinear optimization is similar to the intrinsic camera calibration, with the difference that here the intrinsic parameters are assumed to be known and can remain unchanged. An optimization of the intrinsic parameters in the course of this optimization is also possible. It should be noted that the cameras can be freely aligned with one another as long as their cones of vision intersect and that different cameras (with different intrinsic parameters) can also be used. An example for the extrinsic calibration of a multimodal camera pair consisting of a monochrome camera operating in visual range and an imaging THz scanner is shown in Fig. 5.

5 Calibration of inverse cameras (projectors)

In the case of a camera, the 3D scene is projected onto the 2D image plane, while in the case of a projector a 2D pattern is mapped into the 3D space. Therefore, projectors are often referred to and modeled as inverse cameras. The mathematical theory underlying the projection is the same for a camera and an inverse camera (projector). As a result, the same calibration algorithms can be used. The only challenge when calibrating a projector is to precisely determine the calibration points in a projector image.

This problem can be solved elegantly with a camera that captures the projection of a light pattern on a calibration target. The decoding of the camera image then yields the projector coordinates that are needed for calibration. A complementary Gray code sequence was implemented in the 3D-EasyCalib™ as a universal and widespread method for structured light coding. This pattern can be generated for any projector resolution and robustly decoded in the camera image.

Similar to the procedure for subpixel-accurate measurement of calibration points in camera images, their positions now have to be estimated in projector image coordinates. The idea comes from [8] in which the precise position of the calibration point is computed with a homography between the camera and projector patches around each calibration point. The camera does not have to be calibrated in advance for this. The methods for calibrating a projector with a help of a camera in the 3D-EasyCalib™ thus consist of the detection of calibration points in projector image and combination of intrinsic and extrinsic stereo calibration, which have already been discussed in the previous sections.

Calibrated projectors are often used to capture objects in 3D with structured light (Fig. 6) or for augmented reality, such as projection for the purpose of marking objects in 3D scenes or projecting user interfaces in space (Fig. 9).

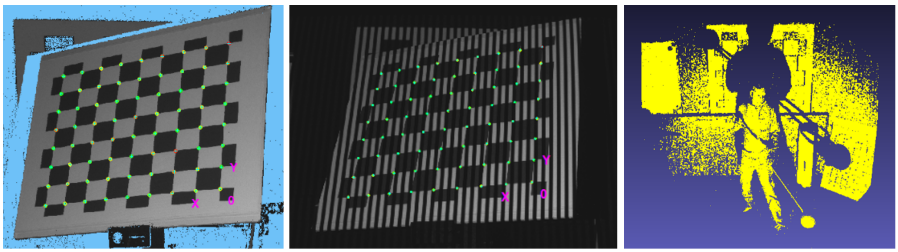

Fig. 6: For the projector calibration, the calibration points in the projector image (left) are determined using Gray code pattern captured with a camera (center). On the right is the 3D reconstruction of a scene with a camera-projector system calibrated in this way.

6 Camera-robot calibration

The calibration of a camera to a robot is always required when a robot has to interact or cooperate with an object that is detected with a camera. A distinction must be made here between whether the camera is moved along with the robot's gripper or whether the camera is stationary and observes the robot from a fixed position in space. In the literature, these two problem classes are represented by the equations $AX = YB$ and $AX = XB$. The transformations A, B, X, Y are matrices of the form $\left[\begin{smallmatrix} R & \boldsymbol{t} \\ \boldsymbol{0}^T & 1 \end{smallmatrix}\right]$. The equations indicate the invariants that are used in order to solve X and Y.

6.1 $AX = YB$: Robot/world- and tool/flange-calibration

To determine the Euclidean transformation between the coordinate systems of a stationary camera and a robot base (Y) as well as the relative position and orientation of the tool to the flange (X), a calibration target is fixed to the end effector of the robot. The target is moved in different positions with the robot and recorded with the camera. The target pose B is automatically determined using the known intrinsic camera parameters from (3). The exact position and orientation of the end effector (Tool Center Point, TCP) A is output by the robot. The equation $AX = YB$ can now be interpreted as a transformation from the calibration target to the robot coordinate system in two different ways, see Fig. 7 top. For each different flange or target pose i, the linear system of equations is put together as $A_iX = YB_i$ and solved in a closed form using the method of Shah [9].

The accuracy of the calibration is influenced not only by the target detection error, but also by the positioning accuracy of the robot. We define the total error (3D error) of the calibration as the Euclidean norm of the differences between the calibration points on the calibration target $\boldsymbol{X_k}$, transformed with A_iX (via TCP) and transformed with YB_i (via camera) $\|(A_iX - YB_i)\boldsymbol{X_k}\|$. This error can be calculated in the 3D-EasyCalib™ for each target pose and is displayed visually similar to the reprojection error, see Fig. 8.

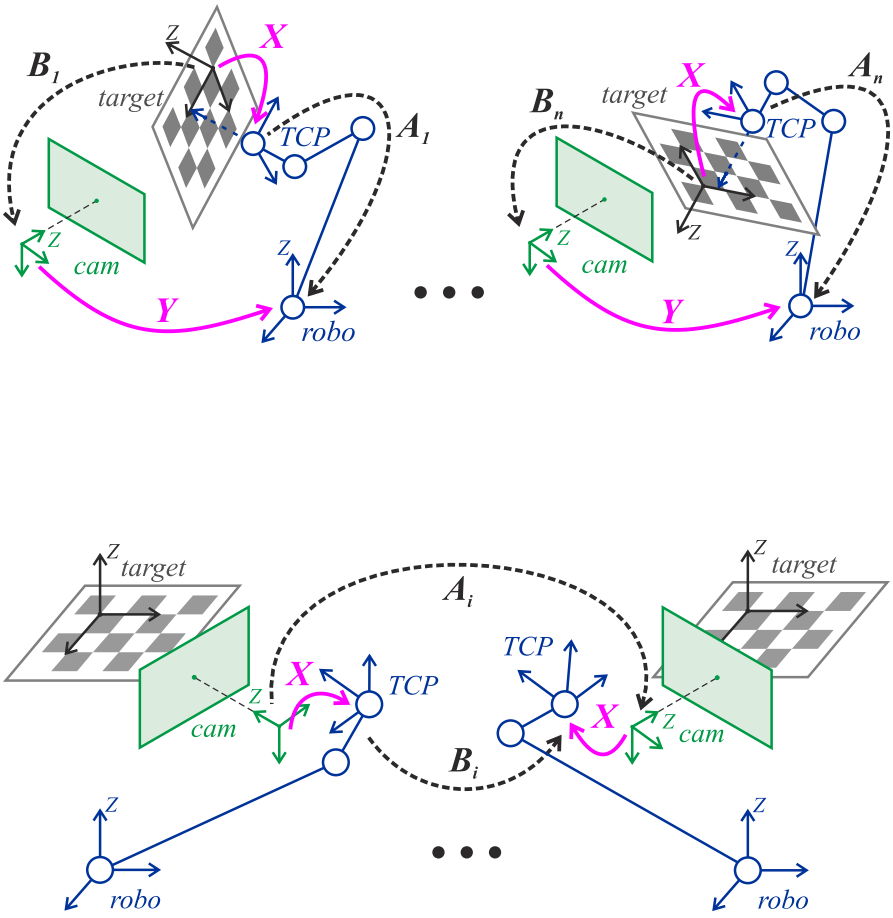

Fig. 7: Symbolic representation of the two problem classes in camera-robot calibration. The robot/world and tool/flange calibration is shown above and the hand/eye calibration is shown below.

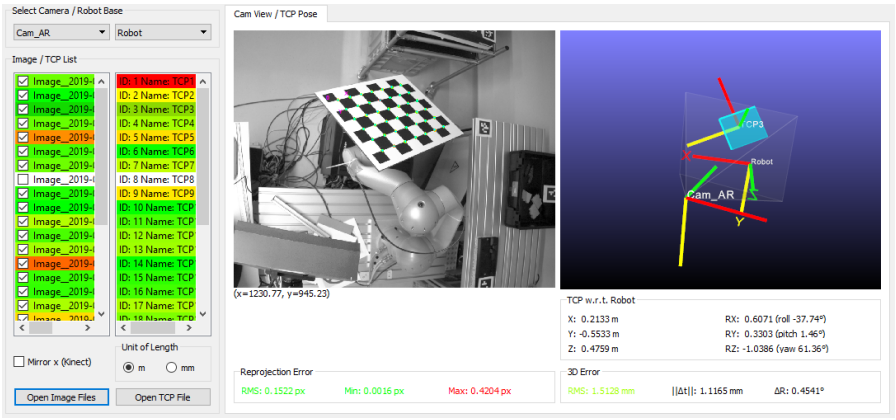

Fig. 8: User interface for the robot/world and tool/flange calibration. The left column shows the individual camera recordings with color-coded reprojection errors. In the column next to it, the TCP poses are marked according to the 3D error. A selected image is shown in the center and the components are shown in 3D on the right.

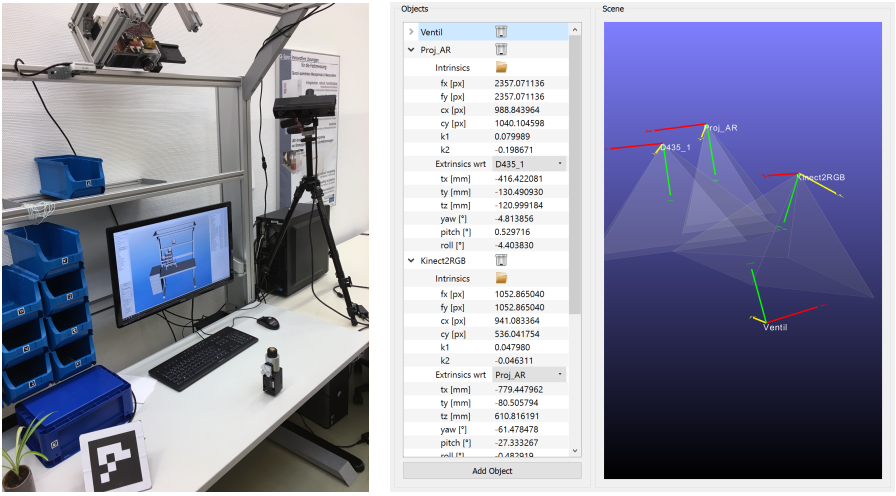

Fig. 9: Demonstrator of a workstation for assembly processes at ZBS e. V. (left). The assembly worker's actions are verified by tracking his hands with several cameras and comparing the information with an assembly process plan. The sensor system consists of a time-of-flight camera for person recognition, three active stereo cameras for component recognition and hand tracking, and a projector for augmentation. The illustration on the right shows the calibrated components of the assembly in the software front end of the 3D-EasyCalib™.

6.2 $AX = XB$: Robot tool to camera calibration (hand/eye)

The problem of hand-eye calibration consists in the computation of a rigid transformation between a camera mounted on the robot actuator and the actuator itself, i.e. the Euclidean transformation X between the camera and the tool coordinate system. The transformation sought is determined with the help of a calibration target that is fixed in space. It is assumed that the robot system knows the exact position and orientation of the end effector. The robot moves to some predefined positions and the camera captures the respective target pose.

The equation $AX = XB$ (see Fig. 7 bottom) represents the fixed position and orientation of the target with respect to the robot for different images. The sought transformation from the camera to the tool coordinate system X is calculated using A and B, which represent transformations between two camera poses and the transformations between two TCP poses, respectively. The comparison of different approaches from the literature with synthetic as well as real data showed that the approach according to Park [10] is the most precise here. It solves the X from $AX = XB$ in closed form. The front end of the 3D-EasyCalib™ for this calibration task corresponds to that of the robot/world and tool/flange calibration.

The best known application of this type of calibration is the “bin picking” scenario. Here a robot picks up randomly placed objects from a container.

7 Conclusion and outlook

The presented applications demonstrate the many possible uses of the modular, universal toolkit 3D-EasyCalib™ (www.zbs-ilmenau.de/3D-EasyCalib) for geometric calibration. So far, different software tools have to be used for various calibration tasks. These often use varying conventions for the calibration parameters and use dissimilar calibration targets.

The 3D-EasyCalib™ combines state-of-the-art methods with ergonomic and intuitive operation and user guidance. For this purpose, tools for the visualization and analysis of errors, the 3D visualization of system components and tutorials were implemented on the basis of the theoretical discussion of calibration problems presented in the article. In this way, non-experts should be able to carry out calibrations quickly and with sufficient accuracy, and experts should be able to set the numerous parameters to precisely analyse the process in terms of the specific applications.

The focus of the upcoming further development of the 3D-EasyCalib™ lies in the improvement of the methods for camera-to-robot calibration with non-linear optimization.

In addition to the classic target-based calibration methods discussed, the 3D-Easy Calib™ includes the scene-based calibration [11]. In this type of calibration, the orientation and position of the cameras is determined using the Manhattan world assumption without the use of calibration targets. The estimation of extrinsic camera parameters from vanishing points is a good alternative or addition to the existing calibration procedures, especially for large urban or industrial workspaces.

8 Acknowledgements

The results presented in this article were created within the framework of the funding program ”twenty20 - Partnership for Innovations” in the joint project ”Ergonomics Assistance Systems for Contactless Human-Machine Operation (EASY COHMO)” funded by the German Federal Ministry of Education and Research.

References

[1] J. Mallon and P. F. Whelan, “Which pattern? Biasing aspects of planar calibration patterns and detection methods,” *Pattern Recognition Letters*, vol. 28, no. 8, pp. 921–930, Jun. 2007.

[2] G. Bradski, “The OpenCV Library,” *Dr. Dobb’s Journal of Software Tools*, 2000.

[3] W. Förstner and E. Gülch, “A fast operator for detection and precise location of distict point, corners and centres of circular features,” in *Proceedings of the ISPRS Conference on Fast Processing of Photogrammetric Data*, 1987, pp. 281–305.

[4] A. Duda and U. Frese, “Accurate detection and localization of checkerboard corners for calibration,” in *29th British Machine Vision Conference, BMVC-29*, Newcastle, UK, Sep. 2018.

[5] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd. New York, NY,USA: Cambridge University Press, 2003.

[6] Z. Zhang, “A flexible new technique for camera calibration,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 11, pp. 1330–1334, Nov. 2000.

[7] J. Bouguet, *Matlab camera calibration toolbox*, 2001.

[8] D. Moreno and G. Taubin, “Simple, accurate, and robust projector-camera calibration,” in *Proceedings of the 2012 Second International Conference on 3D Imaging, Modeling, Processing, Visualization & Transmission*, ser. 3DIMPVT ’12, Washington, DC, USA: IEEE Computer Society, 2012, pp. 464–471.

[9] M. Shah, “Solving the robot-world/hand-eye calibration problem using the kronecker product,” *Journal of Mechanisms and Robotics*, vol. 5, pp. 7–31, 3 Aug. 2013.

[10] F. Park and B. Martin, “Robot sensor calibration: Solving AX = XB on the euclidean group,” *IEEE Transactions on Robotics and Automation*, vol. 10, pp. 717–721, 5 Oct. 1994.

[11] D. Vehar, R. Nestler, and K.-H. Franke, “Präzise Berechnung von Kameraposen in manhattan-welten,” in *21. Anwendungsbezogener Workshop zur Erfassung, Modellierung, Verarbeitung und Auswertung von 3D-Daten, 3D-NordOst*, GFaI Gesellschaft zur Förderung angewandter Informatik e. V., Dec. 2018, pp. 15–24.