A SIMPLE HAND-EYE CALIBRATION METHOD FOR A 3D LASER RANGE SENSOR

Mikko Sallinen, Tapio Heikkilä

VTT Automation

PO Box 13023

FIN-90571 Oulu

FINLAND

Mikko.Sallinen@vtt.fi, Tapio.Heikkila@vtt.fi

<http://www.ele.vtt.fi/aut/>

A simple and a reliable full pose (translation and orientation) calibration for a range sensor is necessary for the measurement applications of industrial robots. In this paper we present a method based on hand-eye-calibration. We estimate the parameters using the Bayesian -form iterative model, which also provides possibilities to estimate error covariances of the pose parameters. In addition to this, we can use the sensor error covariance to estimate to combined uncertainty in e.g. measurement applications.

1. INTRODUCTION

In very flexible production, the programming of the whole robot system should be done off-line to maintain productivity. Flexibility often implies also the use of different types of sensor, for locating the workpieces and guiding the motions when the robot is in contact with the workpiece. To guarantee the quality of the robot work the robot motions should be executable within certain levels of accuracy related to the workpiece surface. We approach these problems by modeling and maintaining estimates for spatial uncertainties for such geometric transformations in which the path accuracy on the workpiece surface depends on. For locating the workpiece in the robot workspace the geometric relationship between the vision sensor and the robot coordinate systems should be determined accurately.

The hand-eye calibration process is typically composed of a set of robot motions with related measurements of a vision sensor. The hand-eye calibration methods can be divided into three different forms, depending of the calibration object and forms of the motion. 1) a simple calibration object, strictly chosen motion (Takahashi & Tomita, 1988), 2) complicate calibration object with free motions (Tsai & Lenz, 1989) and 3) simple calibration object with free motions (Heikkilä et al, 1999). Flexibility and ease of use is best in the case 3) to which our method also belongs. The calibration object we use is a simple plane surface and motions can be freely

chosen. However, to have a good estimate with a low level of uncertainties, a versatile set of samples should be chosen.

This paper consists of the following chapters: in chapter 2 we derive the equations for estimating the parameters and the uncertainties they include. In chapter 3 the experimental results are presented, and in chapter 4 we give our conclusions.

2. EXTERNAL SENSOR CALIBRATION

The sensor we are considering is a laser based 3D range sensor. The sensor is based on the structured light and triangulation and gives a set of 3D coordinates in the sensor space. The sensor uses a light stripe which is created by laser diodes. The measuring part of the sensor includes a CCD-array, with which the laser stripe is detected. One measurement gives 566 3D points.

The measuring phase is carried out by executing several motions to random positions close to the calibration plane, inside the sensor's measuring range. The motions should include as versatile a set of locations and orientations as possible. Selecting the measurement points randomly we assume to cover the solution space reasonably well (practically this seems to work well) though with more emphasis on the creation on the sample set would obviously give better results, especially for a smaller size of sample sets.

The input information for the calibration process are the robot TCP values and sensor range values, in the form of a set of 566 points (x,y,z) in the sensor frame. In addition, there should be rough estimates for the tool pose parameters and the calibration plane parameters. Because of the amount of the data from the sensor we obtain, the algorithm is designed to operate in an iterative manner. Since we apply Bayesian -form estimation, the uncertainties for the model will also be determined. For this, the uncertainties for the robot and sensor have to be determined a priori. The robot is modeled by using the Denavit Hartenberg method (Renders et al, 1991). With this model we can set the noise values based on robot joints and accurate values into the pose parameters. A joint based model gives us a more accurate noise model of the robot and reduces errors coming from the non-linearities.

2.1 Estimation of the model parameters

The estimated parameters are the sensor tool correction, including translation and orientation parameters and translation and orientation of the calibration plane (z, ϕ_x, ϕ_y). According to the type of the sensor, we are able to calibrate the full 6 DOF pose of the sensor. This gives more reliable information about the TCP parameters. The estimated parameters of the sensor, $\bar{R}_{tcp}^{sensor}$ and $\bar{T}_{tcp}^{sensor}$ can be written:

$$\bar{m}_s = [x \quad y \quad z \quad \phi_x \quad \phi_y \quad \phi_z] \quad (1)$$

One measurement (laser stripe) that we obtain from the sensor, which is in the sensor frame, gives i number of point values, e.g.:

$$\bar{p}_i^m = [r_x \quad r_y \quad r_z] \quad (2)$$

where i is the number of point in one measurement, i.e. $i = 1 \dots 566$
 m is the number of measurement, e.g. $m = 1 \dots 25$

The point $\bar{p}_i^m$ is transferred into the calibration plane frame by transforming into the robot TCP frame (using the initial guess of the sensor pose values), then to the robot frame and finally to the calibration plane frame (using the initial estimate of the calibration plane frame).

To estimate the calibration parameters, we need a fitting function to be minimized. When the calibration plane has the form of a plane, the error function in vector form is the distance between the each measured point $\bar{p}_i^m$ from the plane surface, i.e. in equation form:

$$\bar{e}_i = [0 \quad 0 \quad z_i] \quad (3)$$

where z_i is z-coordinate of the i :th point in the calibration plane frame

Figure 1. Coordinate frames of the calibration system

Figure 1 illustrates the coordinate transformations. In the estimation process, each measurement point $\bar{p}_i^m$ is transferred into the calibration plane frame. This transformation includes two estimated coordinate transformations, $\bar{R}_{tcp}^{sensor}$, $\bar{T}_{tcp}^{sensor}$ and $\bar{R}_{robot}^{plane}$, $\bar{T}_{robot}^{plane}$. Because of this transformation, z_i in error vector $\bar{e}_i$ depends non-linearly on the estimated parameters. By minimizing the error function at the same time we can correct the estimated transformations. The set of measurements has the form of a line, i.e. in the point vector $\bar{p}_i^m$ y value in the sensor frame is zero.

The estimates for the sensor tool correction are calculated by constructing the partial derivatives from the error vector $\bar{e}$ in respect of the estimated parameters $\bar{m}_s$. For the sensor calibration we can write using the chain rule:

$$\frac{\partial \bar{e}_i}{\partial \bar{m}_s} = \frac{\partial \bar{e}_i}{\partial \bar{p}_i^m} * \frac{\partial \bar{p}_i^m}{\partial \bar{m}_s} \quad (4)$$

where $\bar{e}_i$ is the error vector, eq. (3)
 $\bar{m}_s$ is the parameter vector including the estimated parameters, eq. (1)

The equation (4) can be written in the detailed form:

$$\frac{\partial \bar{e}_i}{\partial \bar{m}_s} = [0 \quad 0 \quad 1] \begin{pmatrix} \bar{R}_{robot}^{plane} \\ \bar{R}_{robot}^{tcp} \end{pmatrix}^T \bar{R}_{robot}^{tcp} \bar{R}_{tcp}^{sensor} \begin{bmatrix} 1 & 0 & 0 & 0 & r_z & 0 \\ 0 & 1 & 0 & -r_z & 0 & r_x \\ 0 & 0 & 1 & 0 & -r_x & 0 \end{bmatrix} \quad (5)$$

where $r_x \ r_z$ are x and z values of the measured point in the sensor frame
 $\bar{R}_{robot}^{plane} \ \bar{R}_{robot}^{tcp} \ \bar{R}_{tcp}^{sensor}$ are rotation matrices in respect of the coordinate transformations

In the estimation process, we use the Bayesian -form method. The estimation is similar to the observation model of the extended Kalman filter. To the correction vector for the estimated parameters in the iteration step c, we can write:

$$\bar{\Delta}_s = -(\bar{J}_c^T \bar{Q}_c^{-1} \bar{J}_c)^{-1} \bar{J}_c^T \bar{Q}_c^{-1} \bar{e} \quad (6)$$

where $\bar{Q}_c$ is the weight matrix, see below

The corrections are repeated iteratively until the correction becomes zero.

2.2 Uncertainties in the Bayesian model

The method we present here assumes that uncertainties in the measurement process are deterministic, they have some noise value in each approached pose. If we use the Bayesian model recursively, on every recursive phase we add new uncertainties to the estimation model. This is worst-case situation and gives very poor estimates for the error covariance of the system.

By defining the weight matrix $\bar{Q}$, we are able to include the uncertainties of different input parameters in our estimation model. Let $\bar{K}$ be a block diagonal matrix in which $\bar{K}$'s are partial derivatives of error $\bar{e}$ with respect to errors $\bar{m}_s$ that affect measuring. Each $\bar{K}_1, \bar{K}_2, \dots, \bar{K}_N$ includes the partial derivative of one measurement and partial derivative of the sensor tool correction. In the diagonal block one $\bar{K}_N$ represents one block and contains these partial derivatives.

Error derivative of $\bar{e}$ with respect to the measuring errors of the measured values $\bar{m}_s$, we can write:

$$\frac{\partial \bar{e}_i}{\partial \bar{m}_s} = \frac{\partial \bar{e}_i}{\partial \bar{p}_i^m} \frac{\partial \bar{p}_i^m}{\partial \bar{m}_s} \quad (7)$$

The error derivative $\bar{e}_i$ with respect to the measured points is:
 $\partial \bar{e}_i / \partial \bar{p}_i^m = \partial [0 \ 0 \ z] / \partial [x \ y \ z] = [0 \ 0 \ 1]$ and the measured point with respect to the

measured values ($\bar{m}_e = [x \ y \ z \ \phi_x \ \phi_y \ \phi_z \ r_x \ r_z]$, where first six are pose values and r_x, r_z are sensor horizontal and distance values):

$$\frac{\partial \bar{p}_i}{\partial \bar{m}_e} = \left(\bar{R}_{robot}^{plane} \right)^T \bar{R}_{robot}^{tcp} \begin{bmatrix} 1 & 0 & 0 & 0 & z & -y \\ 0 & 1 & 0 & -z & 0 & x \\ 0 & 0 & 1 & y & -x & 0 \end{bmatrix} \bar{R}_{tcp}^{sensor} \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} \quad (8)$$

where $\bar{R}_{robot}^{plane}$, $\bar{R}_{robot}^{tcp}$, $\bar{R}_{tcp}^{sensor}$ are rotation matrices in respect of the coordinate transformations.

The combined error derivative, for robot and sensor uncertainties is:

$$\frac{\partial \bar{e}_i}{\partial \bar{m}_e} = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \left(\bar{R}_{robot}^{plane} \right)^T \bar{R}_{robot}^{tcp} \begin{bmatrix} 1 & 0 & 0 & 0 & z & -y \\ 0 & 1 & 0 & -z & 0 & x \\ 0 & 0 & 1 & y & -x & 0 \end{bmatrix} \bar{R}_{tcp}^{sensor} \begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & 1 \end{bmatrix} \quad (9)$$

where x , y and z are points in the robot wrist frame.

Let $\bar{R}$ be a block diagonal matrix in which $\bar{R}_1, \bar{K}, \bar{R}_N$ include the covariance matrices of the different spatial uncertainty sources of the measuring system. $\bar{R}_1$ includes the variances of the robot pose parameters and the variances of measured point coordinates added by the sensor covariance matrix. The whole matrix $\bar{R}$ is formed by setting the $\bar{R}_1$ matrices to the diagonal block of the matrix $\bar{R}$ as many times as there are measurements in the measurement data. Uncertainties are located in $\bar{R}$ as diagonal blocks.

Covariance matrices $\bar{Q}$ for the noise of the error $\bar{K}_m$ can be calculated:

$$\bar{Q} = \bar{K} \bar{R} \bar{K}^T \quad (10)$$

The covariance matrix for the pose uncertainty can be obtained by multiplying the covariance matrix with the Jacobian matrix, adding the initial estimate of the covariance matrix and by calculating the matrix inversion.

$$\bar{P}_e = (\bar{J}_e^T \bar{Q}_e^{-1} \bar{J}_e)^{-1} \quad (11)$$

The covariance matrix $\bar{P}_e$ describes the uncertainties of the estimated parameters. To combine the obtained estimates and covariances, following equations were used, c.f. (Luo & Min-Hsiung, 1988). The total covariance of the estimated parameters was obtained is:

$$\bar{P}_{tot} = (\bar{P}_c^{-1} + \bar{P}_{tot,c-1}^{-1})^{-1} \quad (12)$$

where $\bar{P}_{c-1}$ is the covariance matrix obtained from the previous iteration step
 $\bar{P}_{tot,c-1}$ is the total uncertainty from the previous iteration step

And the parameters estimate was updated using the following equation:

$$\hat{m} = \bar{P}_{tot} \bar{P}_c^{-1} \bar{m}_{estimate,c} + \bar{P}_{tot} \bar{P}_{tot,c-1}^{-1} \bar{m}_{estimate,c-1} \quad (13)$$

where $\bar{m}_{estimate,c}$ is the estimate of one iteration from the current iteration step
 $\bar{m}_{estimate,c-1}$ is the estimates from previous iteration step

When considering the sensor fusion and related uncertainties, the form of the uncertainty has to take into account. More information on spatial forms see (Heikkilä et al, 1999). The final uncertainties are given in form A (the zero mean homogenous noise transformation follows the nominal transformation).

2.3 Recursive algorithm

To make calibration more flexible and accurate and handle large data sets with point-to-point surface fitting, a recursive algorithm was constructed. This means that when the model parameters once are estimated, reliability and accuracy of the parameters and related uncertainties can be improved by introducing the new data. In the estimation model, the covariance matrix $\bar{P}_{c-1}$ includes the uncertainties of the estimated parameters from the previous iteration step. The recursive algorithm works as follows:

- 1 Set initial value for estimated parameters $\bar{m}_s$ and covariance $\bar{P}_{c-1}$
- 2 Take every n:th point from each measurement pose, in such way that matrices in the calculation of eq. (6) will not become too small but not too large
- 3 Calculate correction $\bar{\Delta}_s$ for the estimated parameters using the eq. (6)
- 4 Update the estimated parameters $\bar{m}_s$ using the correction $\bar{\Delta}_s$
- 5 If corrections $\bar{\Delta}_s$ are close enough to zero, go to the next phase, Otherwise go to step 3
- 6 Now we have estimates for parameters $\bar{m}_s$. Calculate the covariance matrix $\bar{P}_c$ using the eq. (11)
- 7 Do steps 1-6 as long as all the data has been processed. When all the data is processed, there should be n values for estimates $\bar{m}_s$ and n values for respect covariances $\bar{P}_c$.
- 8 Calculate the total covariance $\bar{P}_{tot}$ using the eq. 13 and respect estimates using eq. 13. Continue calculation until all the estimates and covariances have been processed
- 9 The obtained $\bar{P}_{tot}$ and estimates $\hat{m}$ are the error covariance matrix and estimates for the sensor parameters

3. EXPERIMENTAL TESTS

The algorithm was tested with a simulation tool, and also actual tests. The actual tests were carried out using the ABB IRB1400 robot with a MURR Electronics M2D laser range sensor (MEL, 1998).

3.1 Simulation model

A robot workcell model was constructed in a TELEGRIP simulation tool (Deneb, 2000). The cell includes a robot, a sensor and a table where the calibration plane was located. To make calibration easier, a simple user interface was constructed.

The graphical user interface allows the user to select measurement points from the plane surface by simply using the mouse, clicking on the plane surface. When the surface points, called tags, are selected, they are rotated randomly to create a versatile set of data, e.g. there were both large and small surface gradients. Because of the rather large dimensions of the sensor, very large gradients are not allowed. In figure 2 the selection of the surface points is in process. When the calibration points are run, the interface gives the robot TCP point respect with the calibration points. These points can be run with the actual ABB IRB1400 robot.

Figure 2. The TELEGRIP user interface to select the calibration points

3.2 Algorithm tests

The algorithms were tested first by simulations. As expected, the estimation algorithm behaved well and each individual pose estimation was obtained with less than six iteration steps. The reduction of the uncertainties when the amount of the measurements is increasing was simulated.

Considering the first standard deviations of the errors (square roots of the eigenvalues of the covariance matrix), the first 20 iteration phases has the largest impact on the uncertainties of the estimated parameters. By examining the results as a curve, it has a form of a log function which describes the principle of the recursive algorithm.

Respectively we studied how the inaccuracy decreases when the number of measurement (i.e. the number of poses) increases. The results are illustrated in figure 3. In the covariance matrix the eigenvectors are rotated due the non-linearities of the system and the versatility of the set of samples. According to figure 3, a number of 20 measurements gives a level of 0.015 mm in translation and 0.05 degree in rotation in pose uncertainty values as measured in diagonal terms of the error covariance matrix.

Figure 3. Reduction of the uncertainties: upper line (x,y,z) - diagonal values, and lower line rotations around (x,y,z)-axis

In the table 2 the uncertainty values (first standard deviation) for sensor tool correction pose parameters are given.

Table 2. Pose uncertainty in actual tests

Parameter	Uncertainty value [mm]	Parameter	Uncertainty value [mm]
x	0.0161	ϕ_x	0.0264
y	0.0349	ϕ_y	0.0125
z	0.0172	ϕ_z	0.0580

We have observed that, when considering the eigenvectors and eigenvalues of the covariance matrix (there are both translation and rotation values), the result will be distorted due to the different scales of the units. To get more reliable figure about the uncertainties we calculated single point uncertainties (same scale for all parameters) for the have 3D points in the sensor frame. The points are in the left side of the sensor frame, point (-15,0,25), in the middle (0,0,25) and in the rightmost point in the sensor frame (15,0,25).

According to the results in figure 4, we reach the level of 0.015 mm uncertainty for all parameters (x,y,z) after 22 measurement points. From figure. 4 we can see that x and y uncertainties have reached the minimum level already, but the largest change

in uncertainties can be observed in the z value. Considering the z distance is important because z is the distance value in the sensor.

Figure 4. Uncertainties of a point in the calibrated sensor frame, in left side, in the middle and in the right side of the sensor frame, respectively

The negative side of the recursive algorithm is the convergence. In the Bayesian model, there are a weight matrix $\bar{Q}$ and previous error covariance matrix $\bar{P}_{c-1}$ in the correction equation (5). When we use a very accurate initial value of the estimated parameters and the low level error covariance, a reliable correction $\bar{\Delta}_s$ is difficult to find. The calculated correction steps are very small. This is due to the non-linearities of the linearized system. In this kind of situation, the number of the iteration steps may be up to 20-30. However, a result of the correction close to zero will be reached.

3.3 The actual tests

The actual calibration tests were also carried out using the ABB Robotics IRB1400 robot. The noise for the D-H model of the robot, was obtained from actual tests. The joint errors were clearly varying from joint to joint, see table 1.

Table 1. Noise values in robot joints

Joint number	Noise in [degree]	Joint number	Noise in [degree]
Joint 1	0.008	Joint 4	0.0014
Joint 2	0.0010	Joint 5	0.0020
Joint 3	0.0008	Joint 6	0.0025

The sensor manufacturer gives a sensor inaccuracy of 0.1 mm in the x direction and 0.15 mm in the z -direction, which were used as sensor uncertainties. The calibration plate we use is a standard steel plate.

The obtained tool correction, i.e. values of the external parameters are $\bar{m}_s = [-88.432 \ -94.716 \ 217.716 \ -82.977 \ 49.808 \ 88.302]$. Results of the error covariances is actual tests is shown in table 2.

Table 2. Pose uncertainty in actual tests

Parameters	Uncertainty value [mm]	Parameter	Uncertainty value [degree]
x	0.2137	ϕ_x	0.0671
y	0.0393	ϕ_y	0.0231
z	0.0302	ϕ_z	0.8659

The level of uncertainty in actual tests seems to be a little bit higher than in simulation tests. This is due the lack of versatility of the set of samples in actual measurements. Another reason was that the number of calculated points was 283 per stripe, which gives more reliable result from the sensor. However, by increasing the number of measurements, the level of uncertainty will decrease.

4. CONCLUSIONS

In this paper we presented a simple hand-eye calibration algorithm for a laser range sensor. Such a sensor gives a large amount of data even from one single measurement. Using the recursive algorithm, all the obtained data can be processed. In the estimation we use the Bayesian form where we are able to consider the uncertainties of the system. Correspondingly, with a large amount of data, a very accurate calibration result with low level of uncertainties can be achieved. Based on simulations we estimate that we can reach an accuracy level under 0.015 mm and 0.05 degree for orientation parameters.

When considering the point uncertainty in the calibrated sensor frame, i.e. , effects of the hand-eye calibration to a measured 3D point, after 22 measurements, a level of 0.015 mm of uncertainty can be achieved.

5. REFERENCES

1. Chang, Y-L. & Liang P. On recursive calibration of cameras for robot hand-eye systems. IEEE International Conference on Robotics and Automation. Scottsdale, Arizona, 14-19 May 1989. Washington D.C., IEEE Computer Society Press. Vol. II., 1989, Pp. 838-843.
2. Heikkilä T., Sallinen M., Järviluoma M., A robot based surface inspection system with estimation of spatial uncertainties. Proceedings of SPIE - The International Society for Optical Engineering, Intelligent Robots and Computer Vision XVIII: Algorithms, Techniques, and Active Vision. Vol. 3837 (1999), 242 - 253
3. Luo, R.C., Min-Hsiung L., Robot multi-sensor fusion and integration: Optimum estimation of fused sensor data, Proc. Of IEEE Int. Confrence of Robotics and Automation, 1988, pp. 1076-1081.
4. MEL 1998. LASER-Scanner zur Konturerfassung, Linien-Scanner M2D. MEL Mikroelektronik GmbH, Bahnhofstr. 28a, D-85386 Eching/Germany.
5. Renders J.-M., Rossignol E., Becquet M., Hanus R., Kinematic Calibration and Geometrical Identification for Robots. IEEE Transactions on Robotics and Automation, Vol. 7, No. 6, Dec 1991, pp. 721-732.
6. Takahashi H., Tomita F., Self calibration of stereo cameras, Proceedings of the ICCV 1988, pp 123 - 128
7. Tsai R. Y., Lenz R. K., A new technique for fully autonomous and efficient 3-D robotics hand/eye calibration. IEEE Transactions of Robotics and Automation, vol. 5, pp. 345 – 358, June 1989.
8. www.deneb.com/ Debeb, The Digital Manufacturing Company, 2000.