Automatic Calibration of Industrial Robot and 3D Sensors using Real-Time Simulator

Yuji Ninomiya*, Yuta Arita†, Ryodo Tanaka‡, Takeshi Nishida§ and Nicola Ivan Giannoccaro ¶

Department of Mechanical control Engineering
Kyushu Institute of Technology, 1-1 Sensui, Kitakyushu, Fukuoka, 804-8550, Japan
*Email: ninomiya.yuji769@mail.kyutech.jp
§Email: nishida@cntl.kyutech.ac.jp
Universita del Salento, via per Monteroni 73100, Lecce, Italy
¶Email: ivan.giannoccaro@unisalento.it

***Abstract*—We propose an automatic calibration method of a 3D sensor using a real-time simulator. With this method, the relative posture of a real sensor is estimated by obtaining the rigid body coordinate transformation that associates the virtual and real sensors in real-time. Moreover, the results of applying the method to a real system and verifying its effectiveness are shown.**

***Index Terms*—industrial robot, 3D sensor, automatic calibration, real-time simulator**

I. Introduction

Many studies have been conducted to integrate a virtual space prepared in advance with a physical simulator and virtual reality technology to construct a system for advanced industrial robots [1]. To reduce the teaching procedure of industrial robots according to the change in work content, and develop it as a more autonomous and adaptive system, it is important to recognize not only the target objects but also various three-dimensional (3D) relationships of the objects around them. If the recognized information is transmitted to the robot, it can achieve advanced functions such as motion planning, collision determination, and cooperation with the surrounding devices. Furthermore, the authors have proposed a robotic intelligent space (RIS) [2] that updates the virtual space in real-time using a plurality of 3D measurement sensors and a physical simulator. In addition to the functions realized through the above research, the RIS also enables the real-time recognition of the surrounding obstacles, confirmation of the work content achieved, coordination with workers, abnormality detection, and so on. It also acts as a user interface that visualizes the various behaviors of robots and presents them to workers.

In contrast, when constructing an RIS, it is necessary to install a plurality of 3D measurement sensors (e.g., RGB-D cameras) for measuring the surroundings of the work space of the robot; however, when a rearrangement of the robot and a change in the work content will occur, it is necessary to readjust the installation location of the 3D sensor. In this case, in a conventional sensor system, it is necessary to use a special 3D marker and chess board for 3D calibration, and the procedure for recalibrating them is a bottleneck for a realization of the autonomy and adaptability of industrial robots.

Therefore, in this research, we propose a real-time calibration method for a 3D sensor that can correspond to the sensor' s change in position in real time without requiring a special marker placement. Specifically, first, a virtual industrial robot is reproduced with high precision in the simulator based on the angle of each axis obtained from the controller of the industrial robot. At the same time, a virtual 3D measurement sensor is placed in the simulator, and the virtual robot is measured using the virtual sensor. Calibration is then executed through an estimation of the position where the actual sensor is placed by obtaining this measurement data along with the rigid body coordinate transformation matrix of the measurement data of the actual sensor. In this paper, we describe the details of the proposed method and show the results of adapting the proposed method to an actual industrial robot system.

II. Industrial Robot System

A. System configuration

An overview of the system constructed in this research is shown in Fig.1. An industrial robot (MOTOMAN-SIA5F) and a workbench are used, and the work area including the robot is measured using an RGB-D camera (Kinect V2) included near the robot [2]–[4]. In this case, it is assumed that the robot collects articles to be classified into specified transport boxes.

Next, the signal flow of the system is shown in Fig.2. The posture of the robot and 3D measurement data obtained by the RGB-D camera are transmitted to the server computer. In addition, the server computer sends control signals to the robot. Moreover, the server computer reproduces the robots and sensors in the real-time simulator using these signals, and its operation can be confirmed by linking to real machines. The robot operating system (ROS) was used to construct this system.

III. Coordinate system and sensor data

The relationships between the robot coordinate system and the sensor coordinate systems are shown in Fig.1. First, the

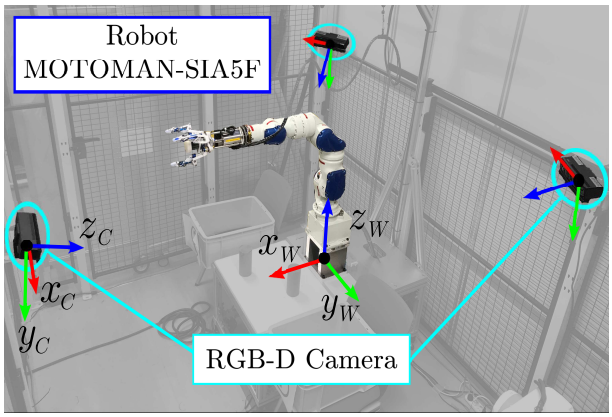

Fig. 1. System overview and coordinate systems of MOTOMAN-SIA5F and Kinect V2.

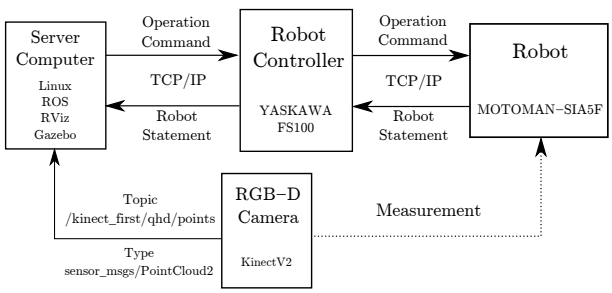

Fig. 2. Signal flow in the system.

coordinate system fixed to the robot is made to coincide with the world coordinate system, as Σ_W.

In addition, the coordinate system of the RGB-D camera is described as Σ_C, and the coordinate system of the virtual sensor is $\Sigma_{\hat{C}}$. The measurement data of the RGB-D camera are a set of 3D points in Euclidean space, and are obtained as point cloud data (PCD).

IV. Calibration Method

A. Method flow

The flow of the proposed calibration method is shown in Fig. 3. These series of processes are executed on the server computer.

First, the robot's posture is received, and a virtual robot is reproduced in a virtual space composed using a real-time simulator. Next, a virtual RGB-D sensor (virtual sensor) is generated in a virtual space with an arbitrary initial position and posture. If the initial position and posture are roughly known, the processing described later can be sped up. In this virtual space, a three-dimensional measurement of the virtual robot is conducted using the virtual sensor, and "virtual measurement data" can be generated. In parallel with this processing, "actual measurement data" from an actual RGB-D sensor (actual sensor) are acquired.

Subsequently, the PCD of the robot is extracted from the virtual measurement data and the actual measurement data. This process was devised to improve the accuracy of alignment by using only the measurement data of a robot with a complicated surface shape, with the exception of a flat work platform surface or the like, from the point group included in the PCD. In addition, at this time, to reduce the number of calculations as necessary, the PCD is reduced through down sampling. After such preprocessing, a rigid body coordinate transformation that associates the virtual measurement data with the actual measurement data is estimated by applying the iterative closest point (ICP) algorithm to these PCDs, and the calibration is finished.

B. Generation of virtual measurement data of virtual robot

To generate the PCD of the virtual robot measured using a virtual sensor, we used the "uniform_sampling function" contained in the reference file "mesh_sampling" of the Point Cloud Library (PCL). With this function, the PCD was generated along the surface shape of the CAD data of an industrial robot.

C. Estimation of rigid body coordinate transformation

The actual measurement data set around the robot are expressed as $A = \{\boldsymbol{a}_i\}_{i=1}^{N}$. Here, $\boldsymbol{a}_i \in \mathbb{R}^3$ represents a 3D point, i is the index of the point, and N is the number of points. In addition, the virtual measurement data are expressed as $B = \{\boldsymbol{b}_j\}_{j=1}^{M}$. Then, a rotation matrix $\boldsymbol{R}$, and a translation vector $\boldsymbol{t}$ that represents the coordinate transform from Σ_C into $\Sigma_{\hat{C}}$, are estimated using the ICP algorithm [6], [7]. This method optimizes a nonlinear error function from a given initial value. The outline of this method is shown below [8].

First, the distance from point $\boldsymbol{a}_i \in A$ to the point set B is defined as follows:

$$d(\boldsymbol{a}_i, B) = \min_{\boldsymbol{b}_j \in B} \|\boldsymbol{b}_j - \boldsymbol{a}_i\| \tag{1}$$

Letting the centers of the actual measurement dataset and the virtual measurement dataset be $\boldsymbol{u}_a$ and $\boldsymbol{u}_b$, respectively, the centroid of each point set matches $\boldsymbol{0}$ through the following operation, and we represent the datasets as A' and B'.

$$A' = \{\boldsymbol{a}'_i | \boldsymbol{a}'_i = \boldsymbol{a}_i - \boldsymbol{u}_a\}, \quad B' = \{\boldsymbol{b}'_j | \boldsymbol{b}'_j = \boldsymbol{b}_i - \boldsymbol{u}_b\} \tag{2}$$

Next, the matrix $\boldsymbol{A}' \in \mathbb{R}^{3\times N}$ is generated by arranging the elements of A'. In addition, the matrix $\boldsymbol{B}' \in \mathbb{R}^{3\times N}$ is

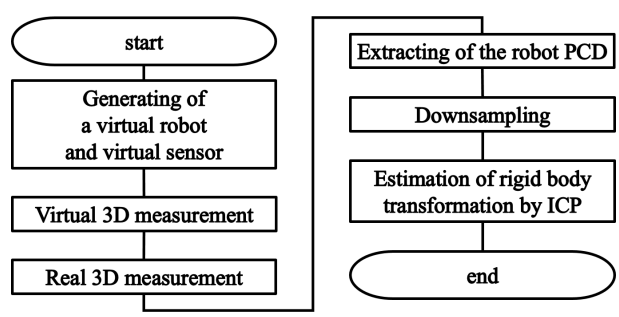

Fig. 3. Calibration flow of proposed method.

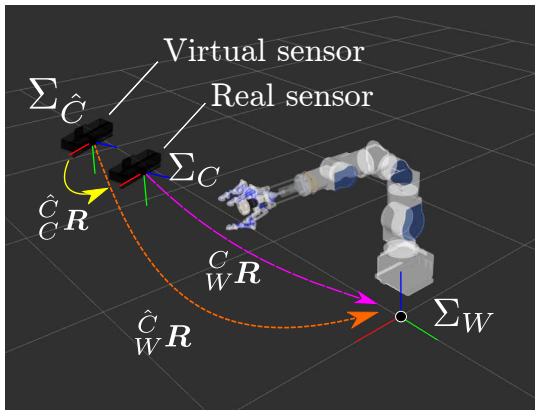

Fig. 4. Relationship of each coordinate system.

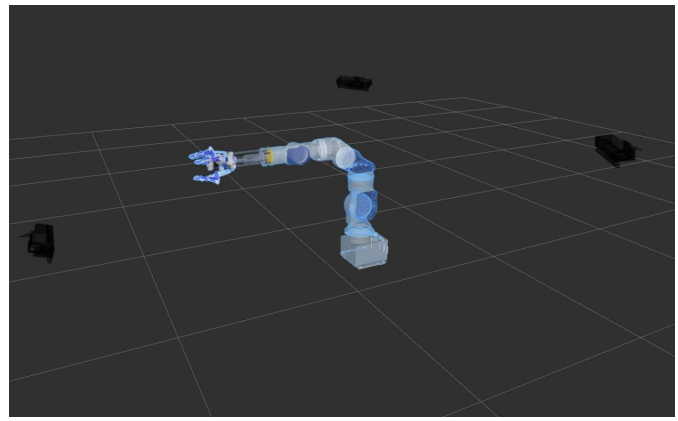

Fig. 5. A state in which the same environment is built in the simulator and the model PCD (blue).

generated by arranging the points $\boldsymbol{b}'_j$ corresponding to the distance between A', as shown in expression (1). The rotation matrix $\boldsymbol{R}$ is obtained through the following singular value decomposition.

$$\boldsymbol{B}'\boldsymbol{A}'^T = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T \tag{3}$$

$$\boldsymbol{R} = \boldsymbol{V}\boldsymbol{S}\boldsymbol{U}^T \tag{4}$$

$$\boldsymbol{S} = \mathrm{diag}(1\ 1\ |\boldsymbol{V}\boldsymbol{U}^{\mathrm{T}}|) \tag{5}$$

Furthermore, the translation vector is obtained as follows:

$$\boldsymbol{t} = \boldsymbol{u}_b - \boldsymbol{R}\boldsymbol{u}_a \tag{6}$$

The distance between the two point sets after a rigid body coordinate transformation is expressed through the following:

$$E(\boldsymbol{R}, \boldsymbol{t}) = \frac{1}{N}\sum_{i=1}^{N} d(\boldsymbol{a}'_i, B') \tag{7}$$

By repeating the procedure and evaluation from equations (3) through (7) through an updating of the point correspondences until the value of $E(\boldsymbol{R}, \boldsymbol{t})$ falls below the threshold value, the estimation of the coordinate transformation is obtained.

As the initial value of the ICP algorithm, a value obtained through a rough actual measurement is used. The correspondence of these rigid body transformations is shown in Fig. 4. When the measured point group A and the model point group B are given, they are related Rigid body coordinate transformation parameters $(\boldsymbol{R}, \boldsymbol{t})$ are estimated by the ICP algorithm.

D. Coordinate transform estimation for calibration

The coordinate transformation parameters are estimated through the following procedure.

1) The actual measurement 3D point set $A \in \Sigma_C$ and the preprocessed virtual measurement point set $B \in \Sigma_{\hat{C}}$ are given.
2) Because $\Sigma_{\hat{C}}$ and Σ_W are in a virtual space, the coordinate transformation $({}^{W}_{\hat{C}}\boldsymbol{R},\ {}^{W}_{\hat{C}}\boldsymbol{t})$ is known.
3) The coordinate transformation $({}^{C}_{\hat{C}}\boldsymbol{R},\ {}^{C}_{\hat{C}}\boldsymbol{t})$ is estimated by applying ICP algorithm to A and B.
4) As the calibration result, the following parameters are obtained.

$${}^{W}_{C}\boldsymbol{R} = {}^{W}_{\hat{C}}\boldsymbol{R} \cdot {}^{C}_{\hat{C}}\boldsymbol{R}^T \tag{8}$$

$${}^{W}_{C}\boldsymbol{t} = {}^{W}_{\hat{C}}\boldsymbol{t} - {}^{C}_{\hat{C}}\boldsymbol{t} \tag{9}$$

V. Experiment

The proposed method was applied to the system shown in Fig. 1, and the calibration was executed. The distance between the Kinect V2 and the robot was set to about 1.340 m. Measurements by the sensor include nominal errors not publicized by the product company.

First, the same environment as shown in Fig. 1 was reproduced in the simulator and visualized as shown in Fig. 5. The PCD shown in blue in the figure represents a PCD generated on the surface of the model CAD data. This point cloud is equal to the integration of data obtained from three virtually installed sensors. Next, only the region of interest is extracted from the actually measured sensor data, and a rigid body coordinate transformation matrix between this and the model data generated earlier is calculated. This situation is shown in Fig 6. The PCD shown in orange in the figure is a point cloud where only the region of interest is extracted from actually measured sensor data, and the PCD indicated by green is the sensor position estimated using this calibration system of the measurement point cloud. Subsequently, we changed the robot's posture and executed the same calibration system. The situation is shown in Fig. 7.

The results of verification of the calibration error for each axis and each angle are shown in Figs. 8 and 9. It was found from these results that the maximum position error was 21.88 mm on the Z axis, and the maximum posture error was 0.023 rad in pitch. Moreover, the average position error rate was 0.007%. From these results, it can be seen that the calibration can be executed through the proposed method according to the change in posture of the robot.

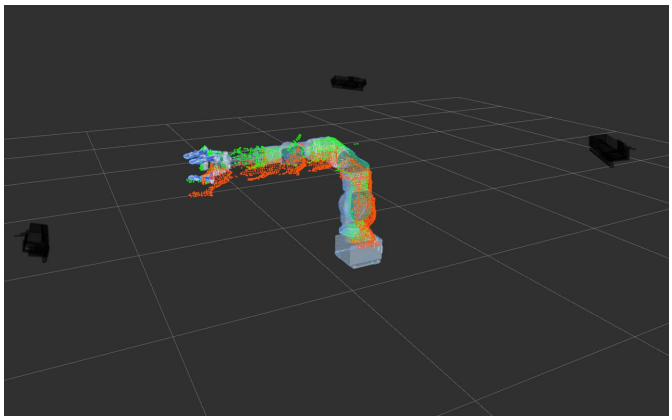

Fig. 6. The PCD before calibration (orange) and after calibration (green).

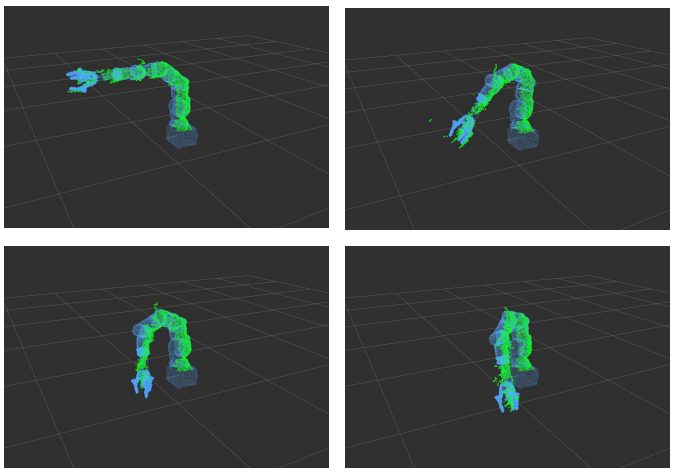

Fig. 7. Measurement of the robot after calibration, and appearance of the simulator.

VI. Conclusion

In this paper, we proposed a real-time calibration method for a 3D sensor that does not require a special marker placement and can respond to changes in the sensor position in real-time. The proposed method is an important technique to solve a bottleneck of the operations of an industrial robot system using virtual reality. In this study, for verification we experimented on a 3D sensor that is not generally applied to industrial use. As future research, we will apply this method to industrial 3D sensors and verify whether we can obtain the accuracy required for general industrial tasks.

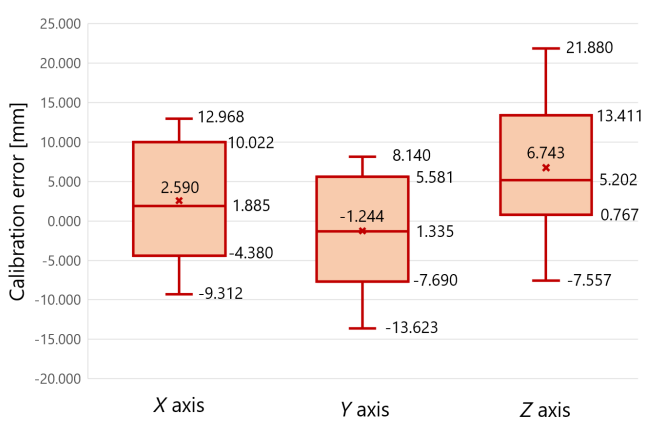

Fig. 8. Calibration error for each axis.

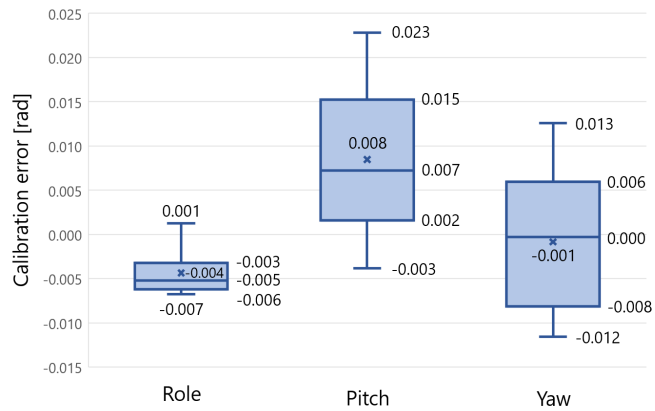

Fig. 9. Calibration error for each angle.

References

[1] E. Manou, G. C. Vosniakos, E. Matsas, "Understanding industrial robot programming by aid of a virtual reality environment," International Journal of Mechanical Engineering Education, 2018.

[2] R. Kabutan, R. Tanaka, S. Oomori, M. Morita, E. Inohira, K. Yoshida, H. Ohtake, T. Nishida, "Development of robotic intelligent space using multiple RGB-D cameras for industrial robots," Proc. of ICT-ROBOT, ThBT3.2, 2016.

[3] R. Tanaka, R. Kabutan, S. Oomori, M. Morita, T. Nishida, "Development of industrial picking system," 4th International Symposium on Applied Engineering and Science, Kyushu Institute of Technology, 17, 2016.

[4] R. Tanaka, R. Kabutan, S. Oomori, M. Morita, T. Nishida, "Development of autonomous picking system with industrial robot," The Society of Instrument and Control Engineers Kyushu Branch, Technical Papers of Annual Meeting, pp. 41–42, 2016.

[5] Zhengyou Zhang, "A Flexible New Technique for Camera Calibration," IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 22, No. 11, pp. 1330–1334, 2000.

[6] Besl, P.J. and Mckay, N.D., "A Method for Registration of 3-D Shapes," IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 14, no. 2, pp. 239–256, 1992.

[7] Takeshi Masuda,"ICP Algorithm," IPSJ SIG Technical Report, vol. 2009-CVIM-168, no. 23, 2009.

[8] Takeshi Nishida and Shuichi Kurogi, "Rotation Matrix Estimation Between Non-Corresponding 3-D Point Sets," JRSJ, vol. 31, no. 6, pp. 624–627, 2013.