IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received December 9, 2020, accepted December 28, 2020, date of publication January 8, 2021, date of current version January 14, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3049964

Efficient TCP Calibration Method for Vision Guided Robots Based on Inherent Constraints of Target Object

ZHIYU YANG, LIANG GONG, (Member, IEEE), AND CHENGLIANG LIU, (Member, IEEE)
School of Mechanical Engineering, Shanghai Jiao Tong University, Shanghai 200240, China

Corresponding author: Liang Gong (gongliang_mi@sjtu.edu.cn)

This work was supported in part by the Project of Science and Technology Commission of Shanghai Municipality under Grant 18391901000, and in part by the National Natural Science Foundation of China under Grant 51775333.

ABSTRACT Tool Center Point (TCP) calibration and target object calibration are essential to guarantee the accuracy of Vision Guided Robot (VGR) systems. After calibration, the robot can know the object's position and orientation from the vision system and then move the TCP to a target point. However, conventional calibration methods are time-consuming and often resort to external tools. We propose a universal method based on the inherent constraints of the target points and use it to simultaneously calibrate the TCP and the target object. In order to obtain the TCP parameters, first, we build a constraint model to TCP by exploiting the target object calibration. By this means, TCP calibration can be combined into the target object calibration, and hereby no external calibration tools are required. Second, we represent this model as an optimization problem of minimizing the reprojection error in the domain of Lie algebra. Third, we solve the numerical problem by the Gauss-Newton algorithm with the perturbation model. Notably, we point out that the universal model can be reduced to a particular simple case when a specific point on the object is available, e.g., a corner of the target object can be recognized by the camera. Here, we present a particular coordinate conversion method to exempt the case from calculating the TCP parameters, which is applicable in a wide range of applications. The practicability and the accuracy of the proposed methods are verified by comparative experiments and numerical simulations. Results show that our method improves TCP calibration efficiency and accuracy by integrating the TCP calibration and object calibration.

INDEX TERMS Vision guided robot, inherent constraint, TCP calibration, target object calibration, simultaneous calibration.

I. NOTATION

For the convenience of description, some symbols and acronyms are defined in Table 1. Without ambiguity, the symbols of homogeneous and non-homogeneous coordinates are not distinguished.

II. INTRODUCTION

A typical Vision Guided Robot (VGR) system [1] is generally composed of cameras, a robot, and a target object. The robot's tool flange is attached to different tools to complete various tasks, such as assembly, picking, and painting. The target object is usually a rigid entity on which, or where, the robot performs its particular tasks. The camera may be mounted around the workspace to provide global visual positioning capabilities. In VGR systems, the camera is used to detect the mutual interactions between the target object and the robot, and to provide visual guidance. Since it is usually tricky for the camera to detect the robot and target object directly, visual fiducial tags such as AprilTag [2] and RUNE-Tag [3] are often employed in practical situations. Figure 1 shows a typical VGR system. Fiducial tags A and B are attached to the robot and the target object, respectively. The camera accurately positions the fiducial tags to derive the mutual position between the robot and the target object.

When the VGR system is performing a specific task, the coordinates of the target point are needed. They vary from time to time in $\mathcal{B}$ but are invariant in $\mathcal{T}_B$. Therefore,

The associate editor coordinating the review of this manuscript and approving it for publication was Yingxiang Liu.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

TABLE 1. Notation of glossary and symbols.

TCP	Tool Center Point
VGR	Vision Guided Robot
EC	Expected Coordinate of a point p in robot coordinate frame, denoted with superscript $'$ defined in Equation 6
$\mathcal{B}$	Robot frame or base frame, which is located on the robot base
$\mathcal{W}$	Wrist frame, which is fixed i to the tool flange
$\mathcal{C}$	Camera frame
$\mathcal{T}_A$	Fiducial tag frame A attached to the tool flange of the robot
$\mathcal{T}_B$	Fiducial tag frame B attached to the target object
$\mathcal{O}$	Work frame attached to the target object
${}^{\mathcal{B}}_{\mathcal{W}}\mathbf{H}$	Homogeneous transformation matrix from $\mathcal{W}$ to $\mathcal{B}$
$[\mathbf{p}]_{\mathcal{W}}$	Coordinate of a point p in $\mathcal{W}$
$[\mathbf{p}_{\mathrm{T}}]_{\mathcal{W}}$	Coordinate of TCP in $\mathcal{W}$

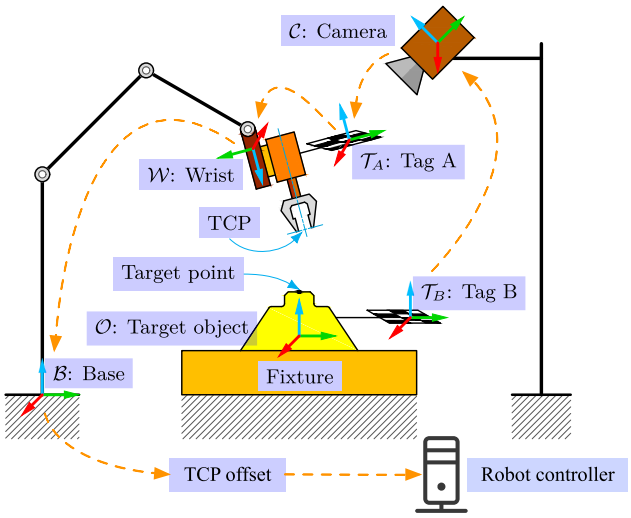


FIGURE 1. A typical VGR system. The orange arrow indicates the coordinate conversion sequence during working. The target point's coordinates are invariant in the Tag B frame, and should be in turn transformed into $\mathcal{C}$, $\mathcal{T}_A$, $\mathcal{W}$, $\mathcal{B}$ coordinate systems. Finally, they are sent to the robot controller after an offset according to the TCP parameters.

it is necessary to convert the target point's coordinates from $\mathcal{T}_B$ to $\mathcal{B}$ when the system is performing a specific task. The orange arrow in Figure 1 indicates the coordinate conversion process. The target point's coordinates are invariant in the Tag B frame, but they should be transformed into $\mathcal{C}$, $\mathcal{T}_A$, $\mathcal{W}$, $\mathcal{B}$ coordinate systems. Finally, the coordinates are sent to the robot controller after an offset, performed according to the TCP parameters. The calibration process of the system is just the opposite, that is, the coordinates from the robot controller should be converted to $\mathcal{T}_B$. The fiducial tags and the tool introduce three invariants, which need to be obtained before performing any task. They are

1) ${}^{\mathcal{W}}_{\mathcal{T}_A}\mathrm{H}$, the transformation relationship between $\mathcal{T}_A$ and $\mathcal{W}$,
2) $[\mathbf{p}_{\mathrm{T}}]_{\mathcal{W}}$, the coordinate of the TCP in $\mathcal{W}$,
3) $[\mathbf{p}_i]_{\mathcal{T}_B}$, the coordinate of the target point p_i in $\mathcal{T}_B$.

In addition, the VGR system calibration involves robot kinematic calibration, which may be further divided into offline [4], [5], and online calibration [6], [7]. Online calibration is conducive to real-time correction during the operation of the system. In particular, online calibration aims at the rapid recovery of the system in response to unexpected situations, such as mechanical arm collisions leading to changes in kinematical parameters. Since robot arms are, in general, calibrated before leaving the factory, in this article, we focus on the three variables ${}^{\mathcal{W}}_{\mathcal{T}_A}\mathrm{H}$, $[\mathbf{p}_{\mathrm{T}}]_{\mathcal{W}}$, and $[\mathbf{p}_i]_{\mathcal{T}_B}$.

Determining ${}^{\mathcal{W}}_{\mathcal{T}_A}\mathrm{H}$ is usually referred to as hand-eye calibration. Many algorithms have been suggested to perform this task. They may be divided in two main categories.

1) *Two-Step Solutions*: Rodrigues-based method by Tsai and Lenz [8], Euclidean-group-based method by Park and Martin [9] and its numerical solution by Gwak *et al.* [10].
2) *Simultaneous Solutions*: Eight-space quaternion approach by Lu and Chou [11], SO(4)-based method by Wu *et al.* [12].

Determining $[\mathbf{p}_{\mathrm{T}}]_{\mathcal{W}}$ is called TCP calibration, which is vital in any robot application, either if it involves offline programming or not. A variety of measurement approaches have been employed for determining TCP, which may be categorized into two groups. The first group consists of those methods using high-accuracy measurement peripherals to track TCP in the world frame. In contrast, in the second group of methods, TCP calibration is done manually by moving the robot and letting the TCP brushes against some particular points. Consequently, these two groups of methods are called non-contact and contract methods, respectively. Several methods belongs to the non-contact ones, and they usually contain a laser-tracker-based or vision-based TCP measurements systems, such as the DynaCal calibration system designed by Cheng [13], the structured-light-based "ten" label detection method of Wang *et al.* [14], the binocular-vision-based welding gun measurement of Wang *et al.* [15]. As for the contact method, Joochim *et al.* [16] and Luo *et al.* [17] have used a reference point. Hong *et al.* [18] have used the geometric constraints between the sphere center and the spherical plane's points. Cakir and Deniz [19] and Han *et al.* [20] have used instead a flat plate as a calibration tool.

Each kind of TCP calibration method has its pros and cons. The non-contact methods have the drawback of being expensive and of limited applications. In particular, they need a particular shape of the tool, which should be precisely positioned by expensive external measurement devices. As a consequence, the contact methods are widely used in the industrial field for their low-cost and high suitability. Nevertheless, the contact methods are usually time-consuming, and the results may fluctuate due to the performance of the robot operator. The operation error may be of the order of 1 mm. Except for high precision hand-eye calibration, the TCP calibration tends to be the most typical cause of VGR system's inaccuracy.

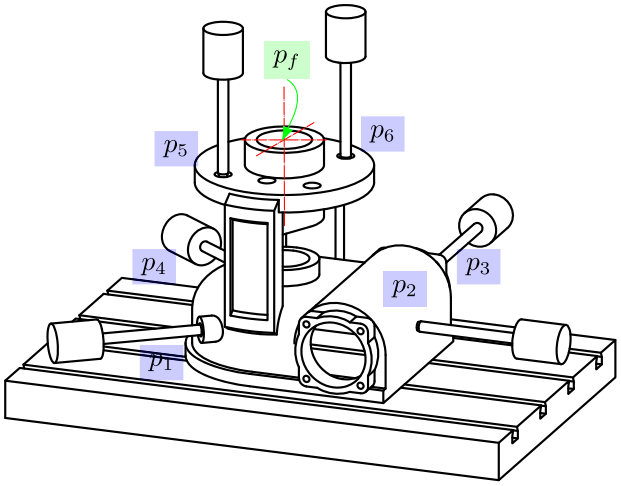


FIGURE 2. An example of the proposed methods. The positions of the target points $p_1 \sim p_6$ may be obtained from CAD data, which are collected during the determination of the target points. p_f is the center of the hole, which may be detected by the camera.

Determining $[\mathbf{p}_i]_{\mathcal{T}_B}$ is called object calibration. This is done after hand-eye calibration and TCP calibration are completed, by manually moving the robot to make it brushes against each target point p_i to derive $[\mathbf{p'}_i]_{\mathcal{B}}$ from the robot controller. Then, one transfers $[\mathbf{p'}_i]_{\mathcal{B}}$ to $[\mathbf{p}_i]_{\mathcal{T}_B}$.

The contribution of this paper is two-fold. First of all, a universal method that uses the inherent constraint of the target points to perform simultaneous calibration of TCP and object is proposed. Notice that in many applications, the relative positions of multiple target points are known and usually defined in CAD data. These relations are used to establish a nonlinear constraint equation for solving TCP parameters. In this paper, we use Lie algebra methods to perform optimization using reprojection error, and implement a Gauss-Newton algorithm based on the perturbation model to compute the numerical solution.

As a second problem, we consider situation in which a specific point on the object can be detected by the camera, and suggest a specific coordinate conversion method based on the relative position of this point and of each target point in the replacement of the concrete TCP parameters. Compared to the first universal method, the second method avoids solving the nonlinear optimization equation and simplifies the calculation.

An application example is shown in Figure 2, where the relative positions of target points $p_1 \sim p_6$ may be obtained from its CAD design. The TCP and object calibration can be done simultaneously according to the first method. If a specific point, e.g. the center of the hole p_f is considered, the second method can be applied. The point p_f can be detected by the camera using the Circle Hough Transform technique. Another common application is for rotatable objects such as tue changers introduced in section V, whose center of rotation can be detected.

Compared with conventional methods reviewed in section III, the main advantages of the proposed methods are

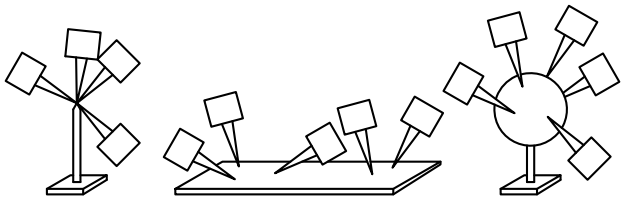


FIGURE 3. Three widely used conventional TCP calibration (of contact) methods based on touching a fixed point, a plane, or a sphere, and using different tool orientations.

1) No external equipment is needed for TCP calibration, leading to high applicability in practical production.
2) No separate TCP calibration steps are needed. TCP calibration and target object calibration are simultaneously done as detailed in section IV. This is more efficient, in general, and it avoids the occurence of large errors caused by long-time manual operation.
3) No nonlinear optimization equation is needed if there is a detectable point on the object, and the coordinate conversion formula may be dramatically simplified, as described in section V.

Finally, simulation and real-world experiments on a fruit picking task and a medical sample changer are conducted and described in section VI to evaluate the proposed particular method's accuracy and time cost. Results indicate that both the universal and the particular methods are competitive in terms of time, and guarantee accurate positioning compared to the conventional method.

III. BACKGROUND KNOWLEDGE

A. CONVENTIONAL TCP CALIBRATION METHOD

The main idea of conventional TCP calibration methods are to move TCP towards some known points that satisfy

$$G\left([\mathbf{p}_{\mathrm{T}}]_{\mathcal{B}}^{(i)}\right) = G\left({}_{\mathcal{W}}^{\mathcal{B}}\mathrm{H}^{(i)}\,[\mathbf{p}_{\mathrm{T}}]_{\mathcal{W}}\right) = 0, \tag{1}$$

where $G(\cdot)$ represents the known constraint of these points. Figure 3 shows three widely used TCP calibration methods based on touching a fixed point, a plane, or a sphere, and using different tool orientations.

For methods based on a fixed point, $G(\cdot)$ may be writtena as

$$G(\mathbf{p}) = \mathbf{p} - [\mathbf{p}_{\mathrm{f}}]_{\mathcal{B}}, \tag{2}$$

where $[\mathbf{p}_{\mathrm{f}}]_{\mathcal{B}}$ is the invariant coordinate of the fixed point in $\mathcal{B}$. For points on the surface of a sphere, $G(\cdot)$ can be expressed as

$$G(\mathbf{p}) = \left\|\mathbf{p} - [\mathbf{p}_{\mathrm{c}}]_{\mathcal{B}}\right\|_2 - r, \tag{3}$$

where $[\mathbf{p}_{\mathrm{c}}]_{\mathcal{B}}$ is the unknown coordinate of the sphere center. and r is the radius of the sphere. For coplanar points, $G(\cdot)$ can be expressed as

$$G(\mathbf{p}) = \begin{bmatrix} a & b & c & d \end{bmatrix}\mathbf{p}, \tag{4}$$

where $[a\ b\ c]^T$ is the normal vector of the plane, and d represents the distance from the origin to the plane.

Equation 1 can be solved directly, or by some optimization model, e.g. using least-square.

B. RUNTIME VISION GUIDE

In a working VGR system, the camera needs to detect the transformation between the robot frame and the object frame in real-time, and then calculate the target point's coordinates in the robot frame. According to Figure 1, when the coordinates of the target point p in $\mathcal{T}_B$ are known and denoted by $[\mathbf{p}]_{\mathcal{T}_B}$, the coordinate of p in $\mathcal{B}$, from now on $[\mathbf{p}]_{\mathcal{B}}$, may be obtained as

$$[\mathbf{p}]_{\mathcal{B}} = {}^{\mathcal{B}}_{\mathcal{W}}\mathrm{H}\, {}^{\mathcal{W}}_{\mathcal{T}_A}\mathrm{H}\, {}^{\mathcal{T}_A}_{\mathcal{C}}\mathrm{H}\, {}^{\mathcal{C}}_{\mathcal{T}_B}\mathrm{H}\, [\mathbf{p}]_{\mathcal{T}_B}. \tag{5}$$

Upon considering the fixed offset between the TCP and the tool flange, let us denote by $[\mathbf{p'}]_{\mathcal{B}}$ the coordinate of $\mathcal{W}$'s origin when the TCP is brushing against p. $[\mathbf{p'}]_{\mathcal{B}}$ can be expressed as

$$[\mathbf{p'}]_{\mathcal{B}} = {}^{\mathcal{B}}_{\mathcal{W}}\mathrm{H} \begin{bmatrix} \mathbf{I} & -[\mathbf{p}_\mathrm{T}]_{\mathcal{W}} \\ \mathbf{0} & 1 \end{bmatrix} {}^{\mathcal{W}}_{\mathcal{T}_A}\mathrm{H}\, {}^{\mathcal{T}_A}_{\mathcal{C}}\mathrm{H}\, {}^{\mathcal{C}}_{\mathcal{T}_B}\mathrm{H}\, [\mathbf{p}]_{\mathcal{T}_B}, \tag{6}$$

where $\mathbf{I}$ is a 3×3 identity matrix. $[\mathbf{p'}]_{\mathcal{B}}$ is called the Expected Coordinate (EC) of p, which can be directly sent to the robot's controller if the tool should perform a task on p.

C. CONVENTIONAL METHOD OF SYSTEM CALIBRATION

The coordinate conversion process involved in system calibration is just the opposite of performing a task (which is shown in Figure 1). Referring to Equation 6, ${}^{\mathcal{T}_A}_{\mathcal{C}}\mathrm{H}$ and ${}^{\mathcal{C}}_{\mathcal{T}_B}\mathrm{H}$ may be measured by the camera, and ${}^{\mathcal{B}}_{\mathcal{W}}\mathrm{H}$ may be generated from the robot's joint space by reading the encoder values. On the other hand, $[\mathbf{p}_\mathrm{T}]_{\mathcal{W}}$, ${}^{\mathcal{W}}_{\mathcal{T}_A}\mathrm{H}$, and $[\mathbf{p}]_{\mathcal{T}_B}$ are three invariant variables that should be determined before production. They are usually referred to as TCP calibration, hand-eye calibration, and object calibration, resepctively.

After TCP and hand-eye calibration, the invariant $[\mathbf{p}]_{\mathcal{T}_B}$ can be obtained by manually moving the robot, letting the TCP to brush the target point p in order to record its EC, denoted as $[\mathbf{p'}]_{\mathcal{B}}$. Subsequently, $[\mathbf{p'}]_{\mathcal{T}_B}$ is calculated according to Equation 6.

IV. THE UNIVERSAL METHOD

In order to obtain the TCP parameters, first, we build a constraint model to TCP by exploiting the object calibration. Object calibration consists in obtaining $[\mathbf{p}_i]_{\mathcal{T}_B}$ by measuring $[\mathbf{p'}_i]_{\mathcal{B}}$ and using Equation 6. If the inherent constraints of the target points p_i are known, e.g., $[\mathbf{p}_i]_{\mathcal{O}}$ is known from CAD data or design drawing, $[\mathbf{p}_\mathrm{T}]_{\mathcal{W}}$ can be calculated from the wrist pose ${}^{\mathcal{B}}_{\mathcal{W}}\mathrm{H}^{(i)}$ when measuring $[\mathbf{p'}_i]_{\mathcal{B}}$ by

$$\mathbf{X}\, {}^{\mathcal{B}}_{\mathcal{W}}\mathrm{H}^{(i)}\, [\mathbf{p}_\mathrm{T}]_{\mathcal{W}} - [\mathbf{p}_i]_{\mathcal{O}} = 0, \tag{7}$$

where $\mathbf{X}$ represents a fixed homogeneous transformation matrix. $[\mathbf{p}_\mathrm{T}]_{\mathcal{W}}$ and $\mathbf{X}$ are not known. By this means, TCP calibration can be combined into the target object calibration, and hereby no external calibration tools are required.

Second, we represent this model as an optimization problem of minimizing the reprojection error in the domain of Lie algebra. Suppose we have m points and $i = 1, 2, \cdots, m$, the optimization may be expressed as

$$\min \sum_{i=1}^{m} \left\| [\mathbf{p}_i]_{\mathcal{O}} - \mathbf{X}\, {}^{\mathcal{B}}_{\mathcal{W}}\mathrm{H}^{(i)}\, [\mathbf{p}_\mathrm{T}]_{\mathcal{W}} \right\|_2^2. \tag{8}$$

Finally, this optimization problem (Equation 8) can be solved by Gauss-Newton method. The iterative formula for the target vector $\boldsymbol{x}$ is

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k - \left(\boldsymbol{J}(\boldsymbol{x})^T \boldsymbol{J}(\boldsymbol{x})\right)^{-1} \boldsymbol{J}(\boldsymbol{x})^T f(\boldsymbol{x}), \tag{9}$$

where the 9-dimensional target vector $\boldsymbol{x}$ can be expressed as

$$\boldsymbol{x} = \begin{bmatrix} \xi \\ [\mathbf{p}_\mathrm{T}]_{\mathcal{W}} \end{bmatrix}_{9\times 1}, \tag{10}$$

where ξ is an element of the Lie algebra of $\mathbf{X}$.

The error function $f(\boldsymbol{x})$ is a $3m$-dimensional vector given by

$$f(\boldsymbol{x}) = \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_m \end{bmatrix}_{3m\times 1}, \tag{11}$$

where e_i is

$$e_i = [\mathbf{p}_i]_{\mathcal{O}} - \exp\left(\xi^\wedge\right)\, {}^{\mathcal{B}}_{\mathcal{W}}\mathrm{H}^{(i)}\, [\mathbf{p}_\mathrm{T}]_{\mathcal{W}}. \tag{12}$$

In order to calculate the Jacobian matrix $\boldsymbol{J}(\boldsymbol{x})$, the partial derivatives with respect to $\mathbf{X}$ and $[\mathbf{p}_\mathrm{T}]_{\mathcal{W}}$ should be computed. Since it is difficult to compute the partial derivatives with respect to $\mathbf{X} \in SE(3)$, we use a disturbance model [21] instead. Partial derivative with respect to $\delta\xi$ and $[\mathbf{p}_\mathrm{T}]_{\mathcal{W}}$ are given by

$$\frac{\partial e_i}{\partial \delta\xi} = -\left[\mathbf{I} - \left[\mathbf{R}_X \left(\mathbf{R}_H^{(i)} [\mathbf{p}_\mathrm{T}]_{\mathcal{W}} + \mathbf{t}_H^{(i)}\right) + \mathbf{t}_X\right]^\wedge\right] \tag{13}$$

and

$$\frac{\partial e_i}{\partial [\mathbf{p}_\mathrm{T}]_{\mathcal{W}}} = -\mathbf{R}_X \mathbf{R}_H^{(i)}, \tag{14}$$

where $\delta\xi$ refers to a tiny disturbance to $\mathbf{X}$. The Jacobian matrix $\boldsymbol{J}(\boldsymbol{x})$ is thus given by

$$\boldsymbol{J}(\boldsymbol{x}) = \begin{bmatrix} \frac{\partial e_1}{\partial \delta\xi} & \frac{\partial e_1}{\partial [\mathbf{p}_\mathrm{T}]_{\mathcal{W}}} \\ \frac{\partial e_2}{\partial \delta\xi} & \frac{\partial e_2}{\partial [\mathbf{p}_\mathrm{T}]_{\mathcal{W}}} \\ \vdots & \vdots \\ \frac{\partial e_m}{\partial \delta\xi} & \frac{\partial e_m}{\partial [\mathbf{p}_\mathrm{T}]_{\mathcal{W}}} \end{bmatrix}. \tag{15}$$

A numerical solution for $[\mathbf{p}_\mathrm{T}]_{\mathcal{W}}$ can be calculated using the iterative formula in Equation 9.

V. THE SPECIAL CASE

The TCP calibration process consists of two steps: manual operations and calculations. The universal method avoids the manual operations by combining them with object calibration. The calculations may be also avoided in some specific situations.

Based on the results of section IV, let us consider a situation where there is a point on the target object which detected by the camera, and the position of this point relative to the target points is known. If the posture of the wrist of the robot is unchanged in the robot frame, TCP parameters can be eliminated in the coordinate conversion formula thus making it possible to avoid the nonlinear optimization equations described in section IV.

We denote data obtained from the calibration process with the superscript (0) and the system running process with superscript (i). The unique point is denote by p_f. According to Equation 6, since ${}^{\mathcal{B}}_{\mathcal{W}}\mathrm{R}$ is unchanged, the EC of the target point p can be expressed as

$$\left[\mathbf{p'}_i\right]^{(1)}_{\mathcal{B}} = \begin{bmatrix}\mathbf{I} & \left[\mathbf{p}_\mathrm{T}\right]_{\mathcal{B}} \\ \mathbf{0} & 1\end{bmatrix} {}^{\mathcal{B}}_{\mathcal{T}_A}\mathrm{H}^{(1)}\, {}^{\mathcal{T}_A}_{\mathcal{T}_B}\mathrm{H}^{(1)} \cdot {}^{\mathcal{T}_B}_{\mathcal{T}_A}\mathrm{H}^{(0)} \left[\mathbf{p}\right]^{(0)}_{\mathcal{T}_A}. \quad (16)$$

Moreover, according to Appendix, $\left[\mathbf{p'}_f\right]^{(0)}_{\mathcal{T}_A}$ can be used to transform Equation 16 into

$$\left[\mathbf{p'}_i\right]^{(1)}_{\mathcal{B}} = \begin{bmatrix}\mathbf{I} & \left[\mathbf{p}_\mathrm{T}\right]_{\mathcal{B}} \\ \mathbf{0} & 1\end{bmatrix} {}^{\mathcal{B}}_{\mathcal{T}_A}\mathrm{H}^{(1)} \cdot \left[\mathbf{Q}\left(\left[\mathbf{p}\right]^{(0)}_{\mathcal{T}_A} - \left[\mathbf{p}_f\right]^{(0)}_{\mathcal{T}_A}\right) + \left[\mathbf{p}_f\right]^{(0)}_{\mathcal{T}_A}\right], \quad (17)$$

where the $\mathbf{Q}$ can be computed by replacing $\mathbf{H}$ and $\mathbf{p}_1$ by

$$\begin{cases}\mathbf{H} = {}^{\mathcal{T}_A}_{\mathcal{T}_B}\mathrm{H}^{(1)}\, {}^{\mathcal{T}_B}_{\mathcal{T}_A}\mathrm{H}^{(0)} \\ \mathbf{p}_1 = \left[\mathbf{p}_f\right]^{(0)}_{\mathcal{T}_A} = {}^{\mathcal{T}_A}_{\mathcal{T}_B}\mathrm{H}^{(0)} \left[\mathbf{p}_f\right]_{\mathcal{T}_B}\end{cases} \quad (18)$$

in Appendix, where $\mathbf{X}$ is the $\mathbf{Q}$ that is required.

Upon simplifying the symbols in Equation 17 and extracting $\left[\mathbf{p'}_i\right]_{\mathcal{B}}$ and $\left[\mathbf{p'}_c\right]_{\mathcal{B}}$, we arrive at

$$\left[\mathbf{p'}_i\right]^{(1)}_{\mathcal{B}} = {}^{\mathcal{B}}_{\mathcal{T}_A}\mathrm{R}\, \mathbf{R}_Q\, {}^{\mathcal{T}_A}_{\mathcal{B}}\mathrm{R} \left(\left[\mathbf{p'}_i\right]_{\mathcal{B}} - \left[\mathbf{p'}_f\right]_{\mathcal{B}}\right) + {}^{\mathcal{B}}_{\mathcal{T}_A}\mathrm{R}\, \mathbf{t}_Q + \left[\mathbf{p'}_f\right]_{\mathcal{B}}. \quad (19)$$

Rather obviously, Equation 19 may be used to calculate $\left[\mathbf{p'}_i\right]^{(i)}_{\mathcal{B}}$ by $\left[\mathbf{p}_f\right]_{\mathcal{T}_B}$ and $\left[\mathbf{p'}_f\right]_{\mathcal{B}}$ without the explicit value of $\left[\mathbf{p}_\mathrm{T}\right]_{\mathcal{W}}$. Moreover, Equation 19 only uses the rotation part ${}^{\mathcal{W}}_{\mathcal{T}_A}\mathrm{R}$ of ${}^{\mathcal{W}}_{\mathcal{T}_A}\mathrm{H}$.

Let us now provide an example of application to obtain $\left[\mathbf{p}_f\right]_{\mathcal{T}_B}$ and $\left[\mathbf{p'}_f\right]_{\mathcal{B}}$: The rotational part's unique feature is that it has the center of rotation, whose position is known in the fixture frame and camera frame. The former is obtained from its CAD data and provides information about $\left[\mathbf{p}_i\right]_{\mathcal{B}}$. The ECs of a series of target points $\left[\mathbf{p'}_i\right]_{\mathcal{B}}$ are obtained under the condition that the ${}^{\mathcal{B}}_{\mathcal{W}}\mathrm{R}$ is kept unchanged. Then the EC of the rotation center $\left[\mathbf{p'}_f\right]_{\mathcal{B}}$ can be calculated accordingly. The position in the camera fram can instead be exploited to obtain the coordinate of the rotatation center in $\mathcal{T}_B$. The method keeps the rotation center and camera fixed with respect to each other, and continuously rotates the object to obtain markers of B images. From those images, a series of ${}^{\mathcal{C}}_{\mathcal{T}_B}\mathrm{H}^{(i)}$ may be obtained. Now, denote by $\left[\mathbf{p}_f\right]_{\mathcal{T}_B}$ the coordinate of the rotation center in $\mathcal{T}_B$: since $\left[\mathbf{p}_f\right]_{\mathcal{C}}$ is invariant, the constraint equation can be expressed as

$${}^{\mathcal{C}}_{\mathcal{T}_B}\mathrm{H}^{(i)} \left[\mathbf{p}_f\right]_{\mathcal{T}_B} = \text{Constant}. \quad (20)$$

By splitting ${\mathbf{H}^{\mathcal{C}}_{\mathcal{T}_B}}^{(i)}$ into a rotation and a translation part, Equation 20 can be transformed into

$$\left({}^{\mathcal{C}}_{\mathcal{T}_B}\mathrm{R}^{(i)} - {}^{\mathcal{C}}_{\mathcal{T}_B}\mathrm{R}^{(1)}\right)\left[\mathbf{p}_f\right]_{\mathcal{T}_B} = -{}^{\mathcal{C}}_{\mathcal{T}_B}\mathrm{t}^{(i)} + {}^{\mathcal{C}}_{\mathcal{T}_B}\mathrm{t}^{(1)}, \quad (21)$$

in which $i > 1$. The matrix form of Equation 21 is

$$\mathbf{R}_\Delta \left[\mathbf{p}_f\right]_{\mathcal{T}_B} = \mathbf{t}_\Delta, \quad (22)$$

where $\mathbf{R}_\Delta$ and $\mathbf{t}_\Delta$ are given by

$$\mathbf{R}_\Delta = \begin{bmatrix} {}^{\mathcal{C}}_{\mathcal{T}_B}\mathrm{R}^{(2)} - {}^{\mathcal{C}}_{\mathcal{T}_B}\mathrm{R}^{(1)} \\ {}^{\mathcal{C}}_{\mathcal{T}_B}\mathrm{R}^{(3)} - {}^{\mathcal{C}}_{\mathcal{T}_B}\mathrm{R}^{(1)} \\ \vdots \\ {}^{\mathcal{C}}_{\mathcal{T}_B}\mathrm{R}^{(n)} - {}^{\mathcal{C}}_{\mathcal{T}_B}\mathrm{R}^{(1)} \end{bmatrix} \quad (23)$$

and

$$\mathbf{t}_\Delta = \begin{bmatrix} -{}^{\mathcal{C}}_{\mathcal{T}_B}\mathrm{t}^{(2)} + {}^{\mathcal{C}}_{\mathcal{T}_B}\mathrm{t}^{(1)} \\ -{}^{\mathcal{C}}_{\mathcal{T}_B}\mathrm{t}^{(3)} + {}^{\mathcal{C}}_{\mathcal{T}_B}\mathrm{t}^{(1)} \\ \vdots \\ -{}^{\mathcal{C}}_{\mathcal{T}_B}\mathrm{t}^{(n)} + {}^{\mathcal{C}}_{\mathcal{T}_B}\mathrm{t}^{(1)} \end{bmatrix}. \quad (24)$$

Then $\left[\mathbf{p}_f\right]_{\mathcal{T}_B}$ can be expressed as

$$\left[\mathbf{p}_f\right]_{\mathcal{T}_B} = \left(\mathbf{R}^T_\Delta \mathbf{R}_\Delta\right)^{-1} \mathbf{R}^T_\Delta \mathbf{t}_\Delta. \quad (25)$$

VI. SIMULATIONS AND EXPERIMENTS

VGR system calibration may be divided into three parts: hand-eye calibration, TCP calibration, and object calibration. Each part can be further divided into two steps: data acquisition and calculation. The difference between our method and the conventional ones is that we can omit the data acquisition step, and instead use data collected in the object calibration to perform the calculations step in TCP calibration. In conventional methods, this step requires extra manual data acquisition. Besides, for target objects with detectable points, we propose a method to complete the system calibration without explicitly solving for the TCP parameters. Then, we avoid the calculation step in TCP calibration.

To verify and assess accuracy and efficiency of our method, we perform two numerical simulations and two comparative experiments. In comparative experiments, we analyze and compare our method to conventional one in terms of TCP calibration(section IV vs. subsection III-A) and system calibration(section V vs. subsection III-C). In practice, when the 2-norm based error between previous and subsequent steps is less than 1×10^{-3}, the iteration (Equation 9) ends.

In the analysis part, the 2-norm error is used to evaluate the accuracy of the method performed. The 2-norm error of TCP parameters can be expressed as

$$\text{error}_{\text{TCP}} = \left\| \left[\mathbf{p}_{\text{T}}\right]''_{\mathcal{W}} - \left[\mathbf{p}_{\text{T}}\right]_{\mathcal{W}} \right\|_2, \tag{26}$$

where $\left[\mathbf{p}_{\text{T}}\right]''_{\mathcal{W}}$ presents the experimental result and $\left[\mathbf{p}_{\text{T}}\right]_{\mathcal{W}}$ presents the ground truth. The 2-norm based position error can be expressed as

$$\text{error}_{\text{position}} = \|\mathbf{p}'' - \mathbf{p}\|_2, \tag{27}$$

where $\mathbf{p}$ presents the ground truth and $\mathbf{p}''$ presents the experimental result.

A. SIMULATIONS

We designed two simulations to:

a) Test the accuracy of the proposed algorithm in deriving TCP parameters in the case of single noise;
b) Analyze the relationship between the accuracy of the proposed algorithm and the number of target points involved in calibration.

1) SIMULATION DATA

We randomly generate data satisfying Equation 7 in order to verify the proposed method. We assume that errors are only introduced during the data acquisition step of object calibration, that is, the measured $\left[\mathbf{p'}_i\right]_{\mathcal{B}}$ are noisy. This is equivalent to add noise to $\left[\mathbf{p}_i\right]_{\mathcal{O}}$ in the Equation 7. The random noise is then added to $\left[\mathbf{p}_i\right]_{\mathcal{O}}$ in order to describe errors due to manual operations.

In general, a normal distribution is suited to describe the positioning errors due to manual operations. On the other hand, those errors have a maximum magnitude and thus we assume that the distribution of manual operation's position error e_i is a truncated normal distribution in $[-0.7, 0.7]$. Its mean μ is 0, and its standard deviation σ is 0.233. Each transformation matrix ${}^{\mathcal{B}}_{\mathcal{W}}\text{H}^{(i)}$ is drawn from a continuous uniform distribution for each degree of freedom.

2) SIMULATION METHOD AND RESULT

At first, in order to test the accuracy of the proposed algorithm at determining the TCP parameters in the presence of single noise, we select eight datasets for each calibration, and perform a grand total of 2000 simulated calibrations. The number of target points is set at 8, that is, m in the Equation 8 is 8. The error distribution form the simulations is shown in Figure 4.

We then analyze the tradeoff between the accuracy of the proposed algorithm and the number of target points involved in the calibration. Since there are 9 unknowns in Equation 7, the minimum number of the target points to get the least-square solution is 4. Using generated test data, we calculate TCP parameters for 4 to 32 target points, and calculations at fixed number of target points are repeated 300 times. Results are shown in Figure 5.

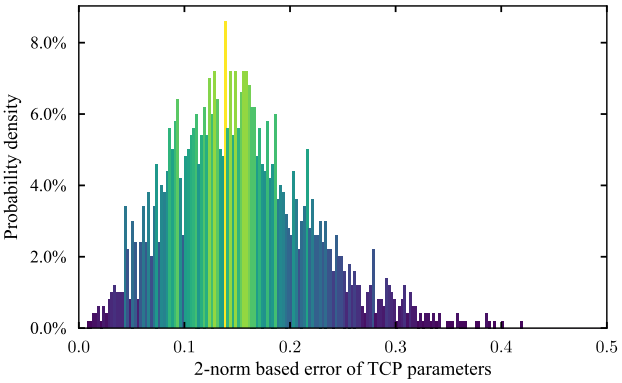


FIGURE 4. The error distribution of the simulation results. 8 target points are used for calculation.

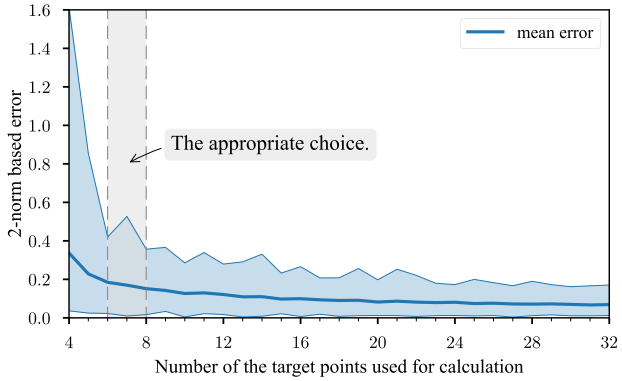


FIGURE 5. The 2-norm error as a function of the number of target points used for the calculation. The shaded area represents the error range, and the solid line is the average error.

3) ANALYSIS

Results in Figure 4 indicate that the most probable calibration error is about 0.1∼0.2 mm, whereas the maximum may exceed 0.4 mm. Compared to the original noise (±0.7 mm), the error of the proposed method is clearly less. This is because, besides the TCP parameters in the target vector $\boldsymbol{x}$ (Equation 10), there are other 6 unknowns. Therefore, the error caused by the original noise is not entirely concentrated on the TCP parameters, which makes the error of TCP parameters only one-third of the original noise.

As expcted Figure 5 also shows that the error decreases with the number of target points used for the calibration. Using 6 to 8 target points appears as the more appropriate choice. In fact, compared to the case of using 4 target points, the average error is nearly halved. Meanwhile, by further increasing the number of target points the improvement is marginal, while the workload of manual operation is linearly arise.

B. EXPERIMENT A: THE COMPARATIVE EXPERIMENT OF TCP CALIBRATION

In this experiment, we implement the proposed universal method (described in section IV) and the conventional TCP calibration method (described in subsection III-A) in a fruit

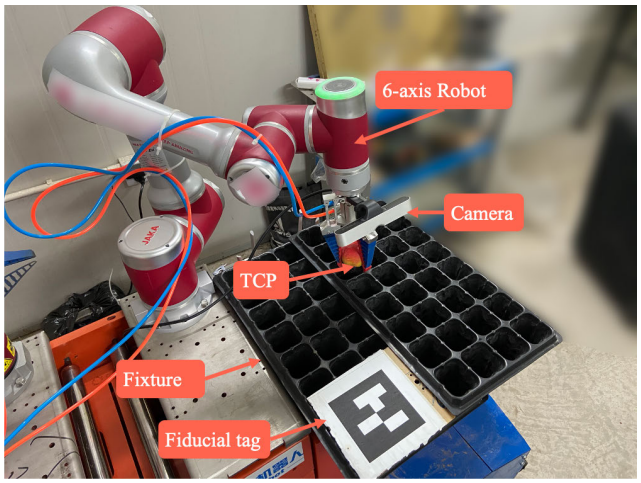


FIGURE 6. Experiment: the System calibration of a 6-axis VGR system.

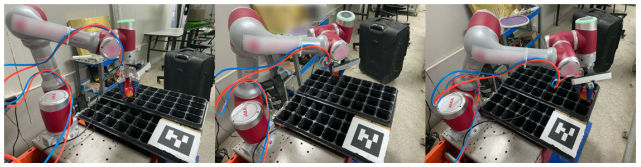


FIGURE 7. TCP calibration and target point calibration are done simultaneously by manually moving the robot, letting the TCP brush the target points.

grasping robot seperately. Then, we compare and analyze the accuracy of the two methods.

1) EXPERIMENTAL METHODS

The VGR system under investigation is shown in Figure 6. The robot model is JAKA Zu® 7 2.0, whose repeatability is 0.03 mm. The distance between holes is 62.5 mm along both directions. The task of this VGR system is to place the fruits in the corresponding holes.

Two methods of TCP calibration are applied. One is the conventional method that is manually moving TCP into a fixed hole using different tool orientations. The other is the proposed universal method, combined with the target point calibration. The scheme partially shown in Figure 7. Each method performed 8 times.

2) RESULT AND ANALYSIS

The experimental results are shown in Table 2. It can be seen from the table that the accuracy of the proposed method is higher than that of the conventional method. This is because of the larger the range of the tool orientation in the calibration process, the greater the accuracy. The conventional method relies on particular external tools to ensure that the TCP repeatedly reaches the same spatial point during the calibration process. However, in this case, there is no such special external tool. Then a fixed target point is used to complete the calibration. Thus, the proposed method that utilizes multiple target points will obtain a great range of tool orientation, thereby improving the accuracy of TCP calibration, and the error is dropped from 2.308 to 1.868.

TABLE 2. The table of TCP parameters calculated by two methods separately.

	x (mm)	y (mm)	z (mm)	error**
ground truth*	0.000	0.000	177.113	-
conventional	0.781	2.137	176.722	**2.308**
proposed	0.331	1.837	177.191	**1.868**

* The value measured in the CAD data of the tool is taken as the ground truth.
** The Euclidean distance from the true value is taken as the error.

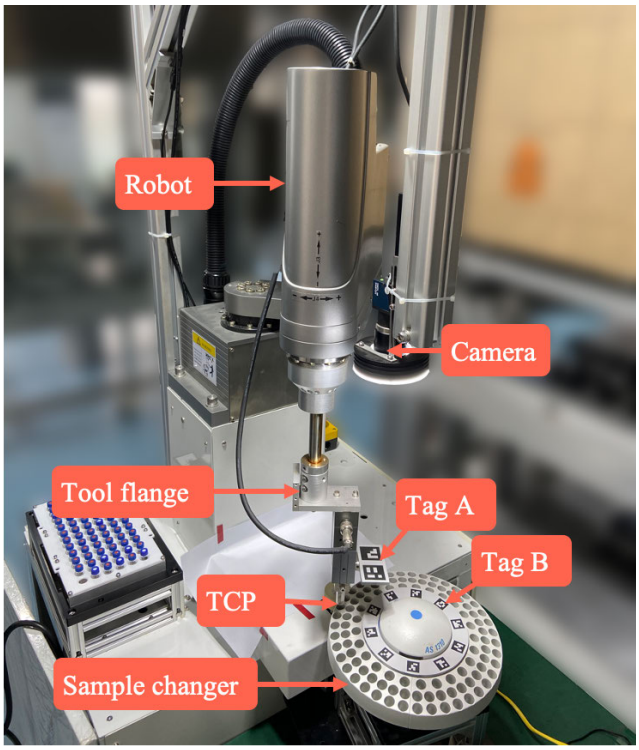


FIGURE 8. Experimental picking and placing for a medical sample changer.

C. EXPERIMENT B: THE COMPARATIVE EXPERIMENT OF SYSTEM CALIBRATION

In this experiment, we performed a system calibration on a sample picking robot on a medical platform. We respectively implement the proposed particular method (described in section V) and the conventional system calibration method (described in subsection III-C). Then, we compare and analyze the system positioning accuracy and overall calibration efficiency of the two methods.

1) EXPERIMENTAL METHOD

The VGR system is shown in Figure 8. It is composed of a robot with four axis, a rotational sample changer, a camera, and a special tool to complete the picking and placing task.

In this VGR system,

1) The repeatability of the robot is 0.04 mm.
2) The camera model is Daheng® MER-1220-32U3C, with 12 million pixels. The linear resolution for fiducial marker is approximately 0.047 mm/pixel.

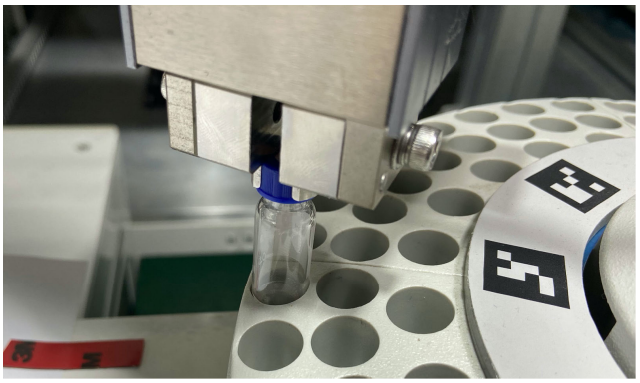


FIGURE 9. Experiment: whether the TCP reaches the target point depends on the precision of manual operations.

3) The printing accuracy of the fiducial marker is ±0.01 mm. The bottom of the sample bottle's outer diameter is 11.6 mm, while the corresponding placement hole's inner diameter on the sample changer is only 12 mm. It means that the positioning accuracy of a successful sample bottle needs to be higher than ±0.2 mm. In a practical situation, since the tool clamping is not entirely rigid, the sample bottle may move slightly. The bottom of the sample bottle has a small arc, so the accuracy in positioning should be about ±0.4 mm. Obviously, a slight collision is taking place when the positioning error is more than ±0.2 mm.

Since the assessment of the TCP is human based, it may cause a large error. As shown in Figure 9, the operator needs to manually control the sample bottle held by the end-tool and put it into a fixed sample hole.

By taking the posture of the wrist constant, the fixed point's manual positioning operation has been repeated eight times, and the measurement results are shown in Figure 10. The minimum enclosing circle diameter is 0.637 mm, which is longer than the physical gap 0.4 mm. This shows that during the manual positioning process, the work object or the sample bottle may be displaced due to the slight touching of the work object, causing larger positioning errors.

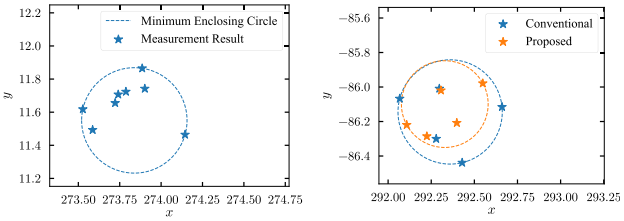
Therefore, in this experimental system, the primary error comes from the manual positioning procedure of the target point, including TCP calibration and target object calibration.

The operation steps of the conventional system calibration method mentioned in section III are as follows.

1) Solve ${}^{\mathcal{W}}_{\mathcal{T}_A}\mathrm{R}$ and ${}^{\mathcal{W}}_{\mathcal{T}_A}\mathrm{t}$.
2) Calibrate TCP by repeated positioning reaching one fixed point.
3) Measure a series of $[\mathbf{p'}_i]_{\mathcal{B}}$ and calculate $[\mathbf{p}_i]_{\mathcal{B}}$ accordingly.

while the operation steps of the proposed method mentioned in section V are as follows.

1) Solve ${}^{\mathcal{W}}_{\mathcal{T}_A}\mathrm{R}$.
2) Keep the gesture unchanged, and measure a series of EC of target points $[\mathbf{p'}_i]_{\mathcal{B}}$. Then calculate the EC of rotation center $[\mathbf{p'}_c]_{\mathcal{B}}$ accordingly.



(a) (b)

FIGURE 10. (a) The repeated positioning error distribution for a single target point. (b) The EC of a pointed target point using 10 different calibration results.

TABLE 3. The EC of a pointed target point using 10 different calibration results.

No.	conventional			proposed		
	x (mm)	y (mm)	error*	x (mm)	y (mm)	error*
1	292.068	-86.067	0.303	292.307	-86.020	0.123
2	292.297	-86.009	0.184	292.108	-86.220	0.223
3	292.429	-86.438	0.265	292.549	-85.979	0.283
4	292.660	-86.115	0.321	292.398	-86.208	0.104
5	292.279	-86.300	0.133	292.225	-86.285	0.170
avg.	292.347	-86.186	**0.241****	292.317	-86.142	**0.181****

* The unbiased estimation of the experimental results of the two methods is used as a reference due to no ground truth, and the 2 norm based error of each experiment is used to measure errors.
** Note that this represents the average of the error, not the error of the average position.

TABLE 4. Time cost analysis of the two calibration methods.

main step	conventional	proposed
TCP calibration (10 poses)	8 min	-
calibrate ${}^{\mathcal{T}_A}_{\mathcal{W}}\mathbf{H}$ or ${}^{\mathcal{T}_A}_{\mathcal{W}}\mathbf{R}$ (8 poses)	2 min	2 min
measure $[\mathbf{p}_c]_{\mathcal{T}_B}$ (10 poses)	-	1.5 min
measure $[\mathbf{p'}_i]_{\mathcal{B}}$ (6 points)	5 min	5 min
total cost	16 min	10 min

The two methods share some calibration steps, so the data obtained for calibrating ${}^{\mathcal{W}}_{\mathcal{T}_A}\mathrm{R}$ and $[\mathbf{p'}_i]_{\mathcal{B}}$ may be used for both.

According to these two methods, the test VGR cell has been calibrated 5 times using each method, and 10 groups of calibration results have been obtained. Using these 10 groups of calibration result to calculate the EC of a pointed target point in different situation, 10 different coordinates are obtained, which are listed in Table 3.

2) RESULTS AND ANALYSIS

In Figure 10, the minimum diameter of enclosing circle is 0.604 mm, while with our method we find 0.503 mm. Since in this case, there is no true value of the EC of the target point, the average is taken as the unbiased estimation value. The 2-norm-based evaluation of errors is shown in Table 3. The time required for manual operation of each step has been also taken into account and shown in Table 4.

Compared to the conventional method, the proposed method only needs the object calibration during system calibration, and it does not require manual operation in TCP calibration process. Also, the proposed method does not need to solve the TCP parameters as in experiment A. The proposed method can exploit data measured for the object calibration to the subsequent system positioning calculations. This difference can be seen in Equation 19 and Equation 6.

Our proposed method's result shows that it is superior to the conventional one in terms of accuracy and efficiency.

1) Calibration accuracy analysis. Table 3 shows that the average error of the traditional method is 0.241, while the average error of the proposed method is 0.181. It can be seen that the performance of the proposed method for system calibration is superior to that of the conventional method, which fluctuates greatly. This is because, in the experiment, the proposed method omits the TCP calibration step, and does not require manual operation to collect data in the TCP calibration step. In this experiment, manual operation is just the largest source of error, and the robot arm positioning error and visual positioning error are far less than the errors caused by manual operations. Therefore, in this experiment, the proposed method clearly improves the calibration accuracy. Additionally, since there is no need to calculate ${}^{\mathcal{W}}_{\mathcal{T}_A}\mathbf{t}$, also this source of error is avoided.
2) Efficiency analysis. Table 4 shows that the new method is more efficient than the conventional method, since it saves from the time-consuming TCP calibration, and improves the calibration efficiency by about 37.5%.

In actual calibration procedures, the calibration accuracy significantly depends on whether the target object is kept static or not during the manual operations. In this respect, our method is superior by reducing calibration steps, which introduces human errors.

VII. CONCLUSION AND OUTLOOK

In this paper, we have focused on VGR systems and proposed a novel efficient method to calibrate the TCP and the target object simultaneously. Our method exploits the target points' inherent constraint to establish the constraint equation used to calculate TCP parameters. Compared to the conventional method, only the object calibration is required. Moreover, a particular method for the rotational object has been proposed, which simplifies the coordinate conversion formula, and avoids calculating the TCP parameters. Since the proposed methods combine the two steps into one, they make it possible to avoid propagation of errors due to long-time manual operations. Results from numerical simulations and experiments indicate that the proposed method ensures the necessary calibration accuracy and dramatically improves the calibration efficiency.

In future work, we consider the design of an online calibration method to correct TCP parameters in real-time, thus ensuring long-term stable operation of the system. Indeed, in some scenarios, the tool is not completely and rigidly connected to the end of the robotic arm, or the tool itself is flexible. In this case, TCP parameters are not fixed, rather they change slightly in time. Besides, we also consider quantitatively the impact of each calibration link on the final operating error of the VGR system, hoping to provide a reference for improving the calibration accuracy of the VGR system.

APPENDIX

Claim: Let a homogeneous transformation matrix $\mathbf{H}$ and a coordinate $\mathbf{p}_1$ are known, and the homogeneous transformation matrix $\mathbf{X}$ which satisfies

$$\mathbf{Hp} = \mathbf{X}\left(\mathbf{p} - \mathbf{p}_1\right) + \mathbf{p}_1 \tag{28}$$

for arbitrary $\mathbf{p}$. Then the $\mathbf{X}$ can be expressed as

$$\mathbf{X} = \begin{bmatrix} \mathbf{R}_H & \mathbf{R}_H\mathbf{p}_1 + \mathbf{t}_H - \mathbf{p}_1 \\ \mathbf{0} & 1 \end{bmatrix}. \tag{29}$$

Proof: Write the homogeneous transformation matrix into the form of a rotation matrix and translation vector. Equation 28 can be then expressed as

$$\mathbf{R}_H\mathbf{p} + \mathbf{t}_H = \mathbf{R}_X\left(\mathbf{p} - \mathbf{p}_1\right) + \mathbf{t}_X + \mathbf{p}_1, \tag{30}$$

which holds for arbitrary $\mathbf{p}$. Notice that $\mathbf{R}_H = \mathbf{R}_X$, then the $\mathbf{X}$ can be expressed as Equation 29. ■

REFERENCES

[1] A. Agrawal, Y. Sun, J. Barnwell, and R. Raskar, "Vision-guided robot system for picking objects by casting shadows," *Int. J. Robot. Res.*, vol. 29, nos. 2–3, pp. 155–173, Feb. 2010.
[2] J. Wang and E. Olson, "AprilTag 2: Efficient and robust fiducial detection," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2016, pp. 4193–4198.
[3] F. Bergamasco, A. Albarelli, E. Rodola, and A. Torsello, "RUNE-tag: A high accuracy fiducial marker with strong occlusion resilience," in *Proc. CVPR*, Jun. 2011, pp. 113–120.
[4] C. Li, Y. Wu, H. Lowe, and Z. Li, "POE-based robot kinematic calibration using axis configuration space and the adjoint error model," *IEEE Trans. Robot.*, vol. 32, no. 5, pp. 1264–1279, Oct. 2016.
[5] T. Sun, B. Lian, S. Yang, and Y. Song, "Kinematic calibration of serial and parallel robots based on finite and instantaneous screw theory," *IEEE Trans. Robot.*, vol. 36, no. 3, pp. 816–834, Jun. 2020.
[6] G. Du and P. Zhang, "Online robot calibration based on vision measurement," *Robot. Comput.-Integr. Manuf.*, vol. 29, no. 6, pp. 484–492, Dec. 2013.
[7] G. Du, Y. Liang, C. Li, P. X. Liu, and D. Li, "Online robot kinematic calibration using hybrid filter with multiple sensors," *IEEE Trans. Instrum. Meas.*, vol. 69, no. 9, pp. 7092–7107, Sep. 2020.
[8] R. Y. Tsai and R. K. Lenz, "A new technique for fully autonomous and efficient 3D robotics hand/eye calibration," *IEEE Trans. Robot. Autom.*, vol. 5, no. 3, pp. 345–358, Jun. 1989.
[9] F. C. Park and B. J. Martin, "Robot sensor calibration: Solving AX=XB on the Euclidean group," *IEEE Trans. Robot. Autom.*, vol. 10, no. 5, pp. 717–721, Oct. 1994.
[10] S. Gwak, J. Kim, and F. C. Park, "Numerical optimization on the Euclidean group with applications to camera calibration," *IEEE Trans. Robot. Autom.*, vol. 19, no. 1, pp. 65–74, Feb. 2003.
[11] Y.-C. Lu and J. C. K. Chou, "Eight-space quaternion approach for robotic hand-eye calibration," in *Proc. 21st Century IEEE Int. Conf. Syst., Man Cybern. Intell. Syst.*, vol. 4, Oct. 1995, pp. 3316–3321.
[12] J. Wu, Y. Sun, M. Wang, and M. Liu, "Hand-eye calibration: 4-D procrustes analysis approach," *IEEE Trans. Instrum. Meas.*, vol. 69, no. 6, pp. 2966–2981, Jun. 2020.

[13] F. S. Cheng, "The method of recovering robot TCP positions in industrial robot application programs," in *Proc. Int. Conf. Mechatronics Autom.*, Aug. 2007, pp. 805–810.
[14] X. Wang, Z. Chen, and Y. Cui, "Study on calibration method of structured light non-contact TCF," *J. Electron. Meas. Instrum.*, vol. 33, no. 1, pp. 135–140, 2019.
[15] C. Wang, Z. Li, X. Wang, Z. Chen, X. Chen, and L. Wang, "Robot TCP self-calibration method based on hand-eye relationship," *Mach. Tool Hydraul.*, vol. 47, no. 17, pp. 6–11, 2019.
[16] C. Joochim, S. Kaewkorn, and A. Kunapinun, "The 9 points calibration using SCARA robot," in *Proc. Res., Invention, Innov. Congr. (RI C)*, Dec. 2019, pp. 1–6.
[17] H. Luo, L. Wang, F. Xiang, W. Ouyang, and P. Wang, "Calibration method of tool coordinate system based on least squares," *Electron. Meas. Technol.*, vol. 43, no. 2, pp. 6–9, 2020.
[18] L. Hong, B. Ji, J. Shen, and X. Yang, "Algorithm research for an industrial robot TCP position calibration," *J. Mach. Des.*, vol. 34, no. 3, pp. 81–85, 2017.
[19] M. Cakir and C. Deniz, "High precise and zero-cost solution for fully automatic industrial robot TCP calibration," *Ind. Robot, Int. J. Robot. Res. Appl.*, vol. 46, no. 5, pp. 650–659, Aug. 2019.
[20] F. Han, P. Li, D. Tan, D. Li, D. Shao, and H. Yan, "Method of using flat plate as calibration tool for robot tool center point calibration," *J. Southwest Jiaotong Univ.*, vol. 55, no. 1, pp. 60–67, 2020.
[21] P. Coelho and U. Nunes, "Lie algebra application to mobile robot control: A tutorial," *Robotica*, vol. 21, no. 5, pp. 483–493, Oct. 2003.

ZHIYU YANG received the B.E. degree in mechanical engineering from Tongji University, Shanghai, China, in 2019. He is currently pursuing the M.S. degree in the area of agricultural robot with the School of Mechanical Engineering, Shanghai Jiao Tong University, Shanghai. His research interests include robot calibration and intelligent agricultural robot.

LIANG GONG (Member, IEEE) received the Ph.D. degree in mechanical engineering from Shanghai Jiao Tong University, China, in 2010. From 2007 to 2008, he was with the Laboratory of Embedded Internet System, Luleå University of Technology, Sweden, as a Visiting Ph.D. Student. From 2010 to 2012, he was a Postdoctoral Fellow with the Laboratory of Power Engineering, Shanghai Jiao Tong University. He has taken charge of multiple projects from the High-Tech Research and Development Program (863 Program), NSF of China, in the field of intelligent control and its applications for agricultural robotics. His research interests include computational intelligence and cognitive science, agricultural robotics, wireless sensor networks, and motor drive power electronics and control.

CHENGLIANG LIU (Member, IEEE) received the Ph.D. degree in mechanical engineering from Southeast University, China, in 1999. He joined the Institute of Robotics, School of Mechanical Engineering, Shanghai Jiao Tong University, as an Assistant Professor. In 2000, he was invited to the University of Cincinnati and the University of Wisconsin, as a Senior Visiting Scholar. He was promoted to a Professor in 2002, taking charge of the Institute of Mechatronics. His research interests include intelligent robot systems, power electronics, MEMS application to precise agriculture network-based monitoring, and GPS/GIS/RS-equipped apparatus/machinery. He was a recipient (twice) of the Award for National Prize of Science and Technology Progress Grade II in China, in 2009 and 2011.