Hand-eye calibration with a remote centre of motion

Krittin Pachtrachai[1], Francisco Vasconcelos[1], George Dwyer[1], Stephen Hailes[2] and Danail Stoyanov[1]

Abstract—In the eye-in-hand robot configuration, hand-eye calibration plays a vital role in completing the link between the robot and camera coordinate systems. Calibration algorithms are mature and provide accurate transformation estimations for an effective camera-robot link but rely on a sufficiently wide range of calibration data to avoid errors and degenerate configurations. This can be difficult in the context of keyhole surgical robots because they are mechanically constrained to move around a remote centre of motion (RCM) which is located at the trocar port. The trocar limits the range of feasible calibration poses that can be obtained and results in ill-conditioned hand-eye constraints. In this paper, we propose a new approach to deal with this problem by incorporating the RCM constraints into the hand-eye formulation. We show that this not only avoids ill-conditioned constraints but is also more accurate than classic hand-eye calibration with a free 6DoF motion, due to solving simpler equations that take advantage of the reduced DoF. We validate our method using simulation to test numerical stability and a physical implementation on an RCM constrained KUKA LBR iiwa 14 R820 equipped with a NanEye stereo camera.

Index Terms—Calibration and Identification; Formal Methods in Robotics and Automation; Computer Vision for Medical Robotics

I. INTRODUCTION

CURRENT surgical practice has shifted towards minimally invasive surgery (MIS) where interventional procedures are performed through small incisions or natural orifices in order to minimise trauma to the patient. Robotic-assisted minimally invasive surgery (RMIS) uses tele-manipulation setups to control the surgical instruments thereby enhancing ergonomics for the surgeon and providing better instrument articulation. The tele-manipulation configuration also provides a suitable platform with potential to introduce computer assisted interventions (CAI) in the operation such as virtual fixtures or overlaying intra- and pre-operative imaging onto the video feed for enhanced visualisation of structural and functional anatomical information underneath the visible tissue surface [1].
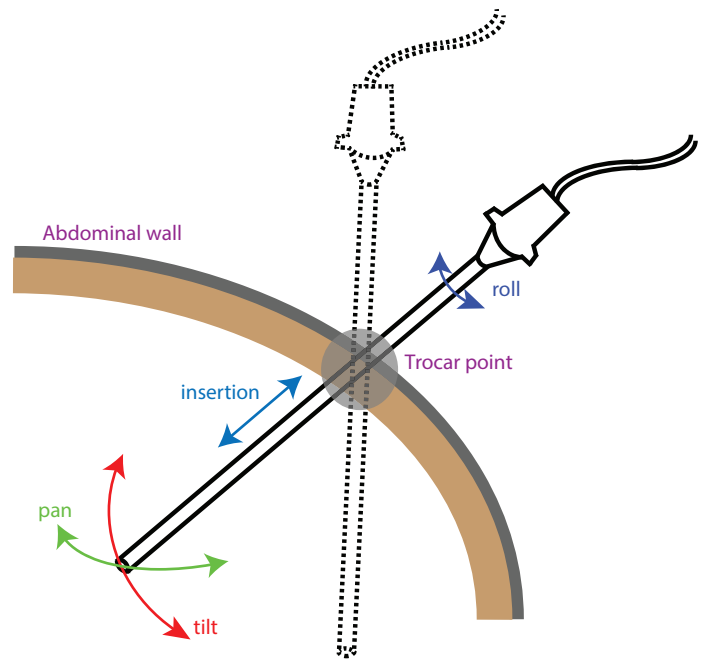
Fig. 1. The schematic shows a magnified version of the type of movement of the camera when being used in RMIS. RCM is denoted at the trocar point to minimise a chance of a robot arm damaging the surrounding tissues [2], [3]. The camera motion is restricted around the RCM and this provides a very small motion range which is not sufficient for a decent calibration.

The practical implementation of CAI systems is critically dependent on performing a hand-eye calibration that links the robot and the surgical camera coordinate systems. Accurate estimation of the hand-eye transformation allows continuous monitoring of a camera motion and the projection of 3D information from the scene into the camera view through forward kinematics [4]. While available hand-eye calibration methods are accurate enough for real-world applications in many domains, this is still not the case in the context of RMIS. One of the main conditions for an accurate hand-eye calibration is to acquire images of a calibration target with a wide range of camera motions that fully explore all 6DoF of the problem. Surgical robots, however, are mechanically constrained to move around a remote centre of motion (RCM) to ensure instruments conform to motion around the trocar entry ports [5] (Fig. 1). This limits the camera motion to only 4DoF (3 in rotation and 1 in translation) and results in ill-conditioned hand-eye constraints. While a possible solution would be to allow a surgical robot to freely move in 6DoF during a calibration phase, this is neither practical nor possible for mechanisms with mechanical RCM implementations [3]. However, as we will show in this paper, it is possible to take advantage of the RCM constraints to improve hand-eye

calibration for such robot configurations.

In this paper, we introduce a new approach to determine the hand-eye transformation in a RCM constrained setting, by first estimating the RCM location in the camera reference frame and then using this information to simplify the classic hand-eye formulation. The advantages and contributions of this approach are:

- Incorporating the RCM position allows the characterisation of robot motions in 4DoF and this changes the hand-eye formulation so that constraints are no longer ill-posed.
- A known RCM position enables formulation of the hand-eye calibration as an absolute pose problem with several stable solutions [6].
- An RCM constrained hand-eye calibration can be more accurate than a classic calibration using free motion, due to the formulation simplifications. This suggests that a more convenient path towards accurate hand-eye calibration in robotic surgery is through correctly modelling its motion constraints rather than allowing the robot to freely move in a calibration phase.

Notation: Matrices are represented by a bold capital letter, e.g. $\mathbf{K}$. Their pseudoinverse is represented by $\mathbf{K}^\dagger$. Bold lowercase letters represent a point or a vector in 3D space in relation to the frame in the subscription, e.g. $\mathbf{p}_{\text{world}}$ represents a point $\mathbf{p}$ in the world frame. The transformation between frame i and frame j is denoted by the 4×4 matrix $^j\mathbf{T}_i$, which consists of the rotation matrix $^j\mathbf{R}_i$ and the translation vector $^j\vec{t_i}$.

II. RELATED WORK

Hand-eye calibration is a classic problem in robotics going back to the 1980s [7] and there is extensive literature on different calibration approaches. The problem is commonly formulated as the homogeneous matrix equation $\mathbf{AX} = \mathbf{XB}$ where $\mathbf{A}, \mathbf{B}$ are the relative transformations in different coordinate frames and $\mathbf{X}$ is the hand-eye transformation [7], [8]. To solve the equation, at least two motions with different axes of rotation are required. Detailed characterisation of camera-robot motions for improving hand-eye calibration accuracy can be found in [9] with considerations for maximising the rotational motion range, minimising the distance between the camera and the calibration target, and minimising the translation motions of the end-effector. Solutions have been proposed to determine $\mathbf{X}$ with different parameterisations e.g. using the special Euclidean groups $SE(3)$ [7], [8], [10], its Lie algebra group $se(3)$ [11]–[13], quaternions [14] and dual quaternions [15], [16], alongside globally optimal solutions for the different parameterisations [17]. Additional constraints, e.g. epipolar constraints from stereo cameras [6] or from Structure-from-Motion (SfM) [18], [19] can be included in the problem and recently probabilistic approaches [20], [21] tackle the challenge of asynchronous data streams from the camera and robot encoders. The motion range, however, remains the same across all of these formulations with the same degeneracy and ill-posed configurations of the original problem setting [8]. While path planning algorithms can ensure the appropriate calibration motions are captured [22], and automated pose
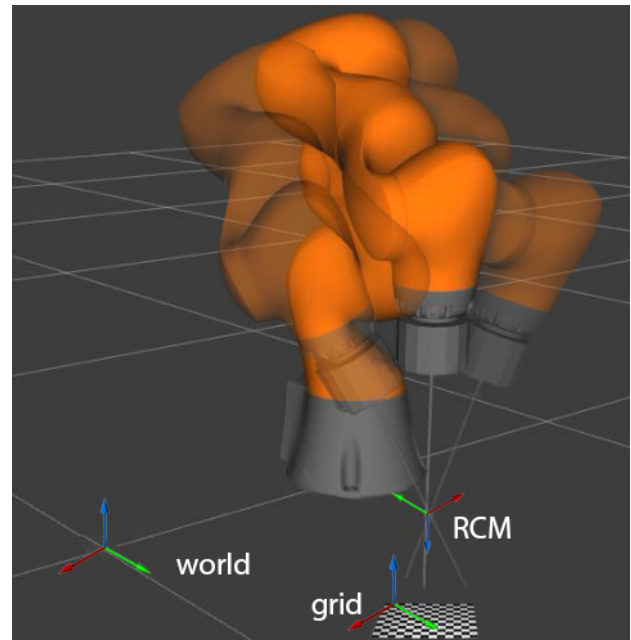

Fig. 2. Example setup for hand-eye calibration shown in simulation using RViz to illustrate the type of the motion around the pre-defined RCM. The coordinate frames in the simulated environment are denoted as shown in the figure. The frame "grid" is usually assigned at the calibration grid, and the frame "world" is assigned as the reference point for the robot pose.

selection methods can guarantee well-conditioned hand-eye constraints [23], [24], when the robot is confined to a limited range the problem remains [9].

The other formulation ($\mathbf{AX} = \mathbf{YB}$) was firstly proposed in [25] which aims to determine both hand-eye and robot-world transformations simultaneously. Several solutions have been proposed to solve the problem [26]–[29]. However, the original paper proves that the formulation contains the similar characteristics as the original hand-eye equation in terms of the rotation estimation. Furthermore, the error propagation on the translation estimation is also shown in [30] that; although the formulation slightly outperforms when the degree of rotation is less than $90°$, the superiority is not significant and it degenerates more quickly than the original equation.

Previous work on the hand-eye problem has also shown that hand-eye calibration of a da Vinci Surgical robot yields significantly larger errors than with other robots mainly due to the strict RCM constraints of the robot motion [13]. While this problem can be attenuated using an EndoWrist surgical tool as a calibration target instead of a static checkerboard [12], [31] and given that it uses RCM constraints, this still heavily relies on accurate visual tool tracking, and thus introduces an additional source of calibration error.

III. PROBLEM FORMULATION

The classic hand-eye calibration procedure has the set-up similar to the one described in Fig. 2. The camera is moved around the calibration grid; collecting several images of the grid with the corresponding robot pose. Finally, the calibration step involves solving the equation below.

$$^{\text{cam},j}\mathbf{A}_{\text{cam},i}\mathbf{X} = \mathbf{X}\,^{\text{robot},j}\mathbf{B}_{\text{robot},i} \qquad (1)$$

where $^{\text{cam},j}\mathbf{A}_{\text{cam},i}$ is the relative motion of the camera at frame cam, i and cam, j and $^{\text{robot},j}\mathbf{B}_{\text{robot},i}$ is the relative motion of the robot arm at frame robot, i and robot, j. Hand-eye calibration determines the rigid transformation $\mathbf{X} = {}^{c}\mathbf{T}_r$ that links the two relative motions. The problem is usually solved by decoupling the rotation component from the whole equation as follows:

$$^{\text{cam},j}\mathbf{R}_{\text{cam},i}\mathbf{R}_X = \mathbf{R}_X {}^{\text{robot},j}\mathbf{R}_{\text{robot},i} \qquad (2)$$

$$^{\text{cam},j}\mathbf{R}_{\text{cam},i}\vec{t}_X + {}^{\text{cam},j}\vec{t}_{\text{cam},i} = \mathbf{R}_X {}^{\text{robot},j}\vec{t}_{\text{robot},i} + \vec{t}_X \qquad (3)$$

The algorithms solve Eq. 2 and use the estimated rotation component to determine the translation component in Eq. 3. This works in the normal setup where the motion is not restricted, but does not work in the RMIS environment. The camera motion in RMIS is restricted around RCM and this allows only small rotational and translational movements between frames which means $^{\text{cam},j}\mathbf{R}_{\text{cam},i}$ and $^{\text{robot},j}\mathbf{R}_{\text{robot},i}$ become very close to identity. Hence, Eq. 2 always holds regardless of the relative motions and cannot be used to solve for rotation. This is later shown in the experiment section where the conventional algorithms always fail at determining the rotation component. Without Eq. 2, the constraint on the rotation component is lost. Therefore, a constraint on the RCM positions $\mathbf{p}_{\text{world}}$ and $\mathbf{p}_{\text{grid}}$ is introduced to solve the problem.

$$^{\text{cam},i}\mathbf{T}_{\text{grid}}\mathbf{p}_{\text{grid}} = \mathbf{X}^{\text{robot},i}\mathbf{T}_{\text{world}}\mathbf{p}_{\text{world}} \qquad (4)$$

The position $\mathbf{p}_{\text{world}}$ is usually pre-defined along the scope as shown in Fig. 1. To formulate the absolute orientation problem, we have to determine the point $\mathbf{p}_{\text{grid}}$.

IV. HAND-EYE CALIBRATION WITH THE REMOTE CENTRE OF MOTION

A. Finding the remote centre of motion in the grid coordinate

To solve the hand-eye problem, we need to find the camera pose that corresponds to each robot pose, i.e. the extrinsic parameters of the camera. This process can be done by solving the homography problem, i.e. estimating the camera pose using the linear decomposition and the initialisation of the camera parameters. Since the camera pose is at the tip of the scope and $\mathbf{p}_{\text{world}}$ is denoted at a point along the scope which is a rigid link, we assume that the lines along the z-axis of the camera poses intersect at the RCM as shown in Fig. 3. This approximation has already been validated in the literature [32].

To determine the point $\mathbf{p}$ of intersection, we have to find the point that minimises the Euclidean distance between itself and its projection on each line. The distance is defined by

$$D = ||(\mathbf{o}_{\text{grid}} - \mathbf{p}) - ((\mathbf{o}_{\text{grid}} - \mathbf{p})^T \mathbf{d}_{\text{grid}})\mathbf{d}_{\text{grid}}|| \qquad (5)$$

where $\mathbf{o}_{\text{grid}}$ is an arbitrary point along the z-axis defined by the frame "cam, i" and $\mathbf{d}_{\text{grid}}$ is a unit vector describing the direction of the axis. Note that every vector is in the grid
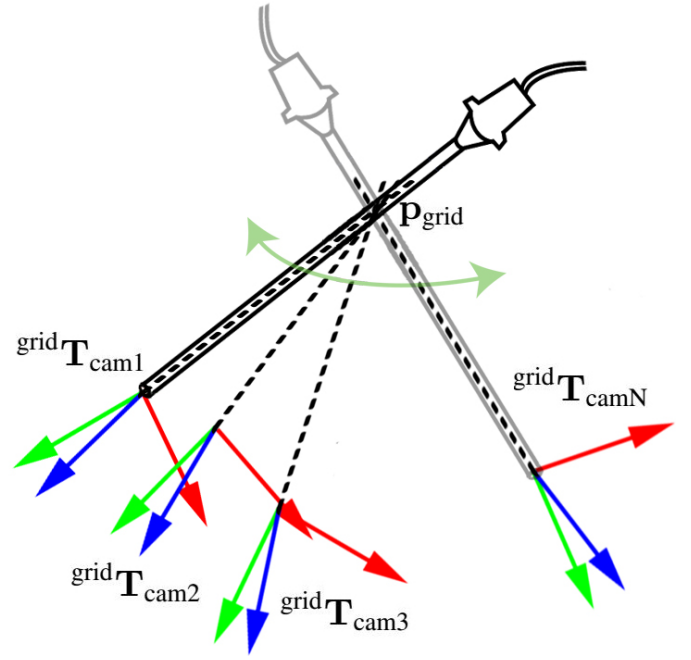

Fig. 3. The schematic shows that the poses of the camera in each frame with respect to the calibration grid coordinate. The blue arrow of each frame denotes the z-axis. The z-axes of the camera poses intersect at the RCM and we can use this information to determine $\mathbf{p}_{\text{grid}}$.

coordinate. Therefore, the total distance from a point $\mathbf{p}$ to a set of lines $(\mathbf{o}_{\text{grid},i}, \mathbf{d}_{\text{grid},i})$ can be represented as

$$D = \sum_{i=1}^{N} w_i ||(\mathbf{o}_{\text{grid},i} - \mathbf{p}) - ((\mathbf{o}_{\text{grid},i} - \mathbf{p})^T \mathbf{d}_{\text{grid},i})\mathbf{d}_{\text{grid},i}||^2$$
$$= \sum_{i=1}^{N} w_i(\mathbf{o}_{\text{grid},i} - \mathbf{p})^T(\mathbf{I} - \mathbf{d}_{\text{grid},i}\mathbf{d}_{\text{grid},i}^T)(\mathbf{o}_{\text{grid},i} - \mathbf{p}) \qquad (6)$$

The parameter w_i is introduced into the equation to put a penalty on some distances more than others, because the estimation of the camera poses always has a re-projection error and the accuracy of the estimation for each pose is not the same. Since the algorithm relies heavily on the RCM position, putting the confidence score on a more accurate pose is crucial to the process.

To minimise this cost function, we have to find the point $\mathbf{p}$ such that $\frac{\partial D}{\partial \mathbf{p}} = \mathbf{0}$. After taking the derivative and re-arranging the equation, we have

$$[\sum_{i=1}^{N} w_i(\mathbf{I} - \mathbf{d}_{\text{grid},i}\mathbf{d}_{\text{grid},i}^T)]\mathbf{p} = \sum_{i=1}^{N} w_i(\mathbf{I} - \mathbf{d}_{\text{grid},i}\mathbf{d}_{\text{grid},i}^T)\mathbf{o}_{\text{grid},i} \qquad (7)$$

By substituting the z-axes and the positions of the camera to $\mathbf{d}_{\text{grid},i}$ and $\mathbf{o}_{\text{grid},i}$, respectively, we can solve Eq. 7 for the RCM in the grid coordinate $\mathbf{p}_{\text{grid}}$ in the least square convention.

After finding the RCM for both cameras, the RCM for the scope can be found by averaging the positions determined from the left and right camera frames,

$$\mathbf{p}_{\text{grid}} = 0.5(\mathbf{p}_{\text{grid},R} + \mathbf{p}_{\text{grid},L}) \qquad (8)$$

Furthermore, the frame at the robot end-effector can be assigned such that the z-axis aligns with the scope as well as the

z-axis of each camera pose to simplify the hand-eye problem to a 4DoF problem; the rotation component of the hand-eye matrix will have the following form:

$$\mathbf{R}_X = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{9}$$

Therefore, according to Eq. 3, the estimation of the translation in the z-axis is no longer affected by the error from the rotation estimation whereas the free motion set-up has a full 6DoF problem and the error from the rotation component subsequently worsens the translation estimation [15]. This behaviour is later demonstrated in Section V that the translation estimation is not consistent and less accurate in comparison to the motion with RCM.

B. Hand-eye calibration using the remote centre of motion

In the previous section, we note that Eq. 2 cannot be used to find the rotation component due to a very limited motion range which leaves only the equation deriving from the translation component. However, Eq. 3 alone cannot be used to solve for both rotation and translation components since it has only 3 equations for 9 parameters. Therefore, we need to use Eq. 4 to create a bigger constraint on the rotation. By solving the homography problem and using forward kinematics, we can determine the transformations $^{\text{cam},i}\mathbf{T}_{\text{grid}}$ and $^{\text{robot},i}\mathbf{T}_{\text{world}}$, respectively. By multiplying the transformations $^{\text{cam},i}\mathbf{T}_{\text{grid}}$ and $^{\text{robot},i}\mathbf{T}_{\text{world}}$ to the points $\mathbf{p}_{\text{grid}}$ and $\mathbf{p}_{\text{world}}$, Eq. 4 can now be written as

$$\mathbf{p}_{\text{cam},i} = \mathbf{X}\mathbf{p}_{\text{robot},i} \tag{10}$$

The above equation can be solved independently from the hand-eye equation for the rotation component and translation component using the absolute orientation algorithm [33]. However, the estimated rotation matrix must conform to Eq. 1 as well and thus it cannot be solely used for solving the hand-eye problem.

According to Eq. 3 and Eq. 4, the estimated rotation matrix must simultaneously yield the translational component $\vec{t}_X$ from both equations and satisfy Eq. 10. Hence, the rotation matrix can be determined from optimising the objective function below using Levenberg-Marquardt,

$$
\begin{aligned}
\vec{r} = \arg\min_{\vec{r}} \Bigg(& \left\| \begin{bmatrix} ^{c2}\mathbf{R}_{c1} - \mathbf{I}_3 \\ ^{c3}\mathbf{R}_{c2} - \mathbf{I}_3 \\ \vdots \end{bmatrix}^{\dagger} \begin{bmatrix} \mathbf{R}_X\, ^{r2}\vec{t}_{r1} - \, ^{c2}\vec{t}_{c1} \\ \mathbf{R}_X\, ^{r3}\vec{t}_{r2} - \, ^{c3}\vec{t}_{c2} \\ \vdots \end{bmatrix} \right. \\
& \left. - (\bar{\mathbf{p}}_{\text{cam}} - \mathbf{R}_X\bar{\mathbf{p}}_{\text{robot}}) \right\| + \sum_{i=1}^{N} \left\| (\mathbf{p}_{\text{cam},i} - \bar{\mathbf{p}}_{\text{cam}}) \right. \\
& \left. - \mathbf{R}_X(\mathbf{p}_{\text{robot},i} - \bar{\mathbf{p}}_{\text{robot}}) \right\| \Bigg)
\end{aligned}
\tag{11}
$$

where $\vec{r}$ is the Rodrigues representation of the rotation matrix $\mathbf{R}_X$, and $\bar{\mathbf{p}}_{\text{cam}}$ and $\bar{\mathbf{p}}_{\text{robot}}$ are the average positions of the RCM in the camera frame and robot frame, respectively. Formulating the objective function in this way allows the two error metrics to have the same unit and thus can be jointly optimised [6].

Moreover, according to the assumption in the previous section, the initial guess of vector $\vec{r}$ can be chosen such that

the z-axis of $^{\text{world}}\mathbf{T}_{\text{cam},i}$ is parallel or anti-parallel with that of $^{\text{world}}\mathbf{T}_{\text{robot},i}$, i.e. $\vec{r}_{\text{init}} = [0, 0, \delta]$ where δ is a small degree of rotation ($\approx 1°$). The problem can then be optimised by a bounded non-linear optimisation that allows some deviation from the alignment of the two axes. In our set-up, we set the threshold at $5°$. Note that the z-axis does not necessarily point out of the end-effector. This depends on the frame assignment at the end-effector which can be arbitrary [34] and results in different initial value $\vec{r}$ and optimisation interval.

To solve for the translation component, we simply stack up the matrices from Eq. 3 and 10 and solve the equation in the least square convention,

$$
\begin{bmatrix} ^{c2}\mathbf{R}_{c1} - \mathbf{I}_3 \\ ^{c3}\mathbf{R}_{c2} - \mathbf{I}_3 \\ \vdots \\ \mathbf{I}_3 \\ \mathbf{I}_3 \\ \vdots \end{bmatrix} \vec{t}_X = \begin{bmatrix} \mathbf{R}_X\, ^{r2}\vec{t}_{r1} - \, ^{c2}\vec{t}_{c1} \\ \mathbf{R}_X\, ^{r3}\vec{t}_{r2} - \, ^{c3}\vec{t}_{c2} \\ \vdots \\ \mathbf{p}_{\text{cam},1} - \mathbf{R}_X\mathbf{p}_{\text{robot},1} \\ \mathbf{p}_{\text{cam},2} - \mathbf{R}_X\mathbf{p}_{\text{robot},2} \\ \vdots \end{bmatrix}
\tag{12}
$$

Note that the indices on the components of the relative transformations can be shuffled to maximise the possible pairwise combination to increase the accuracy as long as the robot's and the camera's are corresponding to each other. The number of possible pairwise combinations is $\binom{n}{2}$.

V. EXPERIMENTS AND RESULTS

The proposed method is validated in this section by comparing its calibration performance with the existing algorithms in [8], [15], [29] and the absolute orientation algorithm [33], noted as TSAI, IDQ, REPROJ and ABSOR respectively. For TSAI, IDQ and ABSOR, We also add the non-linear optimisation at the end of these methods to refine the calibration using dual quaternion parametrisation. Such an objective function is described in [17]. Let $\hat{\mathbf{a}}, \hat{\mathbf{b}}$ and $\hat{\mathbf{q}_X}$ be the dual quaternion of $\mathbf{A}, \mathbf{B}$ and $\mathbf{X}$ and $\otimes$ is the dual quaternion multiplication, the equation can be written as:

$$f(\hat{\mathbf{q}_X}) = \sum_{i=1}^{N} ||\hat{\mathbf{a}}_i \otimes \hat{\mathbf{q}_X} - \hat{\mathbf{q}_X} \otimes \hat{\mathbf{b}}_i||^2 \tag{13}$$

In the literature regarding hand-eye calibration, the validation is done by checking how the algorithm performs under varying noise, motion range and number of motions. In this work, we do not analyse performance for a varying number of motions since its effect in our RCM constrained setting is equivalent to what has been already extensively observed in the free motion calibration case [13], i.e. a higher number of motions improves calibration.

A. Experiments with simulated data

The simulated data is generated by creating a loop of transformations between the calibration grid, the camera frame, the robot arm, and the world frame. Since the robot motion is not random, we define the pre-defined RCM in the simulation and command the robot to move in a spiral motion around the RCM as shown in Fig. 2. This generates 192 robot motions
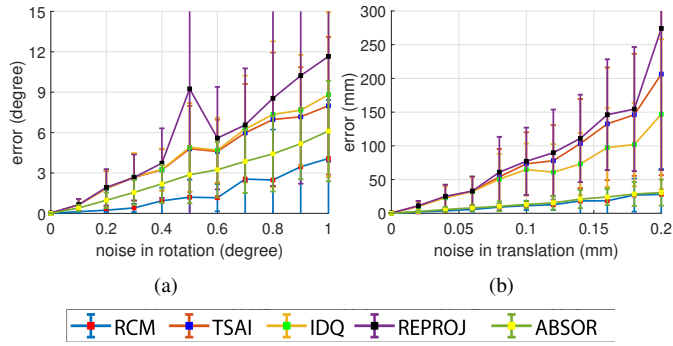
Fig. 4. The comparison of the calibration performance with the increasing noise in both robot and camera motions. (a) Error in the rotation estimation (b) Error in the translation estimation.
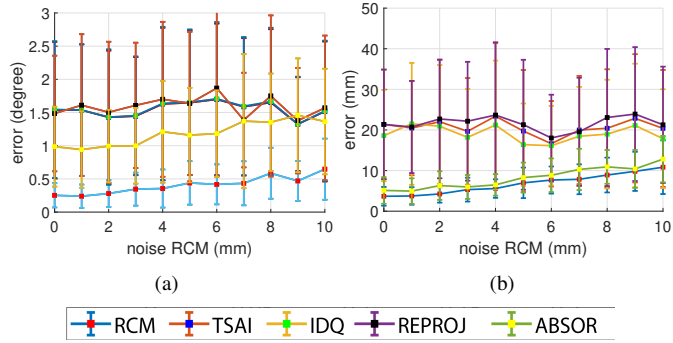

Fig. 5. The comparison of the calibration performance with the increasing noise in the RCM position. The noise in robot and camera motions are kept constant in this experiment. (a) Error in the rotation estimation (b) Error in the translation estimation.
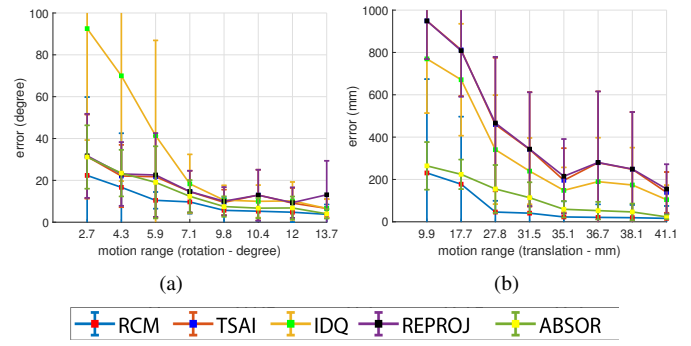

Fig. 6. The comparison of the calibration performance with varying motion range. The noise in robot and camera motions are kept constant in this experiment. (a) Error in the rotation estimation (b) Error in the translation estimation.

around the RCM. The hand-eye matrix is assumed to be a 180 degrees rotation in the x-axis and a translation along the z-axis and the grid is simulated to be right under the scope. The loop can be completed by chaining all the transformations together.

$$^{\mathrm{grid}}\mathbf{T}_{\mathrm{cam},i} = {}^{\mathrm{grid}}\mathbf{T}_{\mathrm{world}}\,{}^{\mathrm{world}}\mathbf{T}_{\mathrm{robot},i}\big({}^{\mathrm{cam}}\mathbf{T}_{\mathrm{robot},i}\big)^{-1} \quad (14)$$

Gaussian noise is then added to the transformations before feeding to the hand-eye algorithm. Noise of 3 mm in RCM position is also added as well to simulate the uncertainty when estimating the RCM in the calibration grid coordinate.

The comparison is run between the RCM method, the classic hand-eye solution [8] and the state-of-the-art dual quaternion solution [15] in different experimental setups by varying the intensity of Gaussian noise, the noise in the RCM estimation and motion range. The accuracy is determined by comparing the estimated transformation with the ground truth of the hand-eye matrix. For each set of simulation parameters, the experiment is run for 100 times.

1) Gaussian noise: The noise in this experiment is increased from 0 mm to 0.2 mm in translation and 0 degree to 1 degree in rotation, while the noise in the RCM position is kept constant at 3 mm. The range of this noise is to simulate the noise in the positioning system of the KUKA robot arm. Fig. 4(a) and 4(b) show that the RCM method outperforms the other four methods when noise is present in the motions. This shows that including Eq. 4 into the calibration creates a constraint on the hand-eye problem which increases the

calibration accuracy and is better than solving the problem using only the absolute orientation algorithm.

2) Noise in RCM: In this experiment, the noise in the robot and camera motions are kept constant at 0.05 mm in translation and 0.25 degrees in rotation whilst increasing the noise in the RCM position from 0 mm to 10 mm. Fig. 5(a) and 5(b) show that the RCM method can provide a better calibration accuracy than the other approaches even with noise in the RCM position. The result presented here is similar to the scenario with the real data where noise is present in every input. If the noise in the RCM is increased further, the proposed method is likely to give a very high calibration error whereas this does not significantly affect the calibration performance of TSAI, IDQ and REPROJ because they do not depend on the RCM. The proposed method and ABSOR, on the other hand, have an increasing calibration error as the noise in RCM is increased, but RCM method can still outperform the ABSOR method.

3) Motion range: The noise intensity in the motions are kept at 0.05 mm in translation and 0.25 degrees in rotation but only a segment of the total motion is fed into the calibration function to vary the motion range. As shown in Fig. 6(a) and 6(b), the RCM method can estimate the hand-eye matrix when the motion range is around 10 mm in translation and 2.7 degrees in rotation whereas the other algorithms fail in the experimental setting. The RCM method still outperforms the others even with increasing motion range. This indicates that the RCM constraint can overcome the problem of restricted motion and it is also applicable in the hand-eye calibration, regardless of the motion range. This simulation experiment shows that our RCM method achieves accurate calibrations in cases where the motion range is ill-posed for other classic hand-eye methods. Although the calibration error still does not satisfy the calibration accuracy that medical robots can operate on [35], the results show the potential of solving the hand-eye problem in such set-up that the original methods and formulations cannot perform.

B. Experiments with real data

The experiment is set up by attaching the scope onto the flange of the KUKA LBR iiwa 14 R820 as shown in Fig. 7 and
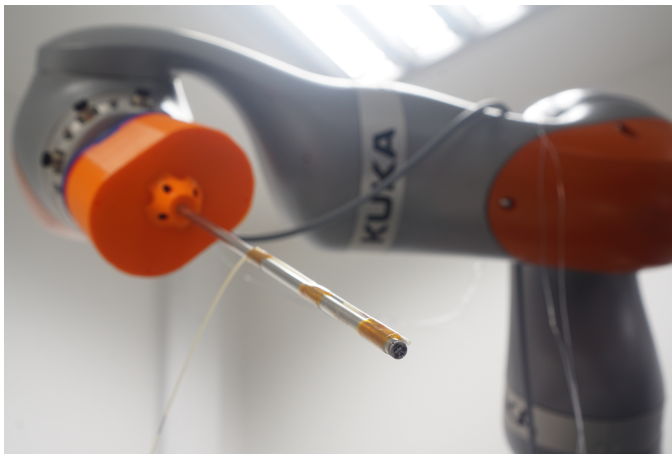
Fig. 7. Experimental setup for capturing data. The specially designed scope with the NanEye stereo camera is mounted on the flange of the KUKA arm. The scope is 420 mm in length. The RCM in the world frame $\mathbf{p}_{\text{world}}$ is pre-defined at a point $[860\text{mm}, -400\text{mm}, 150\text{mm}]^T$ in the world frame which is along the scope. The robot is then commanded to take several captures of the calibration grid at the maximum possible motion range around the fixed RCM.
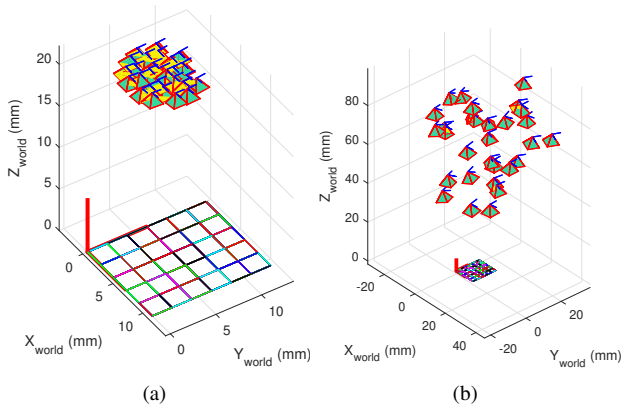

Fig. 8. The camera poses in both experimental setups. (a) The camera motion is restricted around the RCM (b) The camera can freely move around the calibration grid.
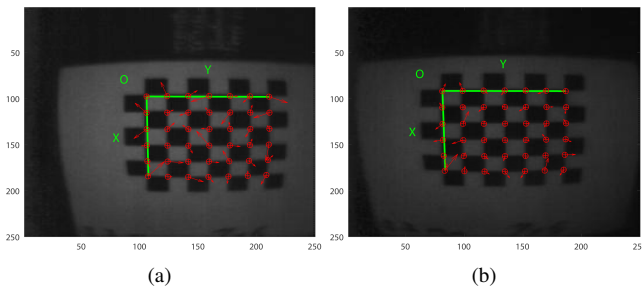

Fig. 9. Images captured by the NanEye stereo camera. The re-projected points are marked as red dots and the grid origin is labelled by the green axes with the z-axis pointing out. (a) Image from the left camera (total re-projection error = 9.6452 pixels) (b) Image from the right camera (total re-projection error = 14.3659 pixels)

the robot is then commanded to move around the RCM with a pre-defined motion to capture several images of the calibration grid. The grid dimension is 7×6 and the size of each block is 2.16 mm. The camera pose is detected and estimated by the

Camera Calibration toolbox.[1] The example of images captured by a NanEye camera is shown in Fig. 9(a)-9(b).

We acquired two sets of poses for the hand-eye calibration. A first one where the motions are constrained by a RCM, and a second where free motions with 6 DoF were performed (Fig. 8). We compare our RCM approach using the RCM constrained motions against the classic methods using both motions sets. The camera poses in both setups are shown in Fig. 8(a)-8(b). It is evident that the RCM-free setup has a wider motion range than the RCM setup.

In the experiment, we use three metrics to validate the performance of each algorithm, the error in rotation, the residue in the hand-eye equation and the error in re-projecting the RCM. The ground truth of the hand-eye matrix is not known in the real experiment, and we cannot use Eq. 2 to estimate the rotation error since the equation always holds for the case of restricted motion range. Therefore, we have to validate the rotation estimation by referring to how coordinate frames are assigned in the robot workspace. In the setup, the z-axis of the world and the grid are always parallel to each other and we can use this setup to evaluate how well the estimated hand-eye transformation recovers the z-axis of $^{\text{world}}\vec{z}_{\text{grid}}$.

$$\text{rotation error} = ||\arccos([0, 0, 1] \cdot {}^{\text{world}}\vec{z}_{\text{grid}})|| \qquad (15)$$

where $^{\text{world}}\vec{z}_{\text{grid}}$ is the third column of the transformation $^{\text{world}}\mathbf{T}_{\text{grid}}$.

Although the residue of the matrix product $(\mathbf{AX})^{-1}(\mathbf{XB})$ does not directly represent the error in the translation estimation, it should suggest how close the estimated solution is to the optimal solution in this formulation. Therefore, together with the error in re-projecting the RCM position, we can infer the overall calibration performance as a combination of these metrics.

For both setups, we collect 30 different camera and robot poses and randomly choose 16 poses as input to each calibration method. After the calibration, we use the unused poses and the estimated hand-eye matrix to compute these three metrics and compare the calibration performance. This process is repeated for 50 times to get a distribution of the error.

Fig. 10(a)-10(c) show the calibration performance of each algorithm (with and without the RCM) in terms of rotation estimation, residue in the hand-eye equation, and error in projecting the RCM position between two coordinates. When the motion is constrained around the RCM position, the other four algorithms clearly fail in calibrating the matrix and give a very high error in comparison to the proposed method. A very high error in rotation shown in Fig. 10(a) indicates that the rotation component of the estimated hand-eye matrix is not correct and cannot be used to recover the correct transformation $^{\text{world}}\vec{z}_{\text{grid}}$. Although the error yielded by REPROJ and ABSOR are comparable to RCM, the translation component is severely worsened by the error in the rotation component which is not the case in RCM method where the z-component is not affected.

Despite having a comparable residue in the hand-eye equation, the consistency in re-projecting the RCM position using
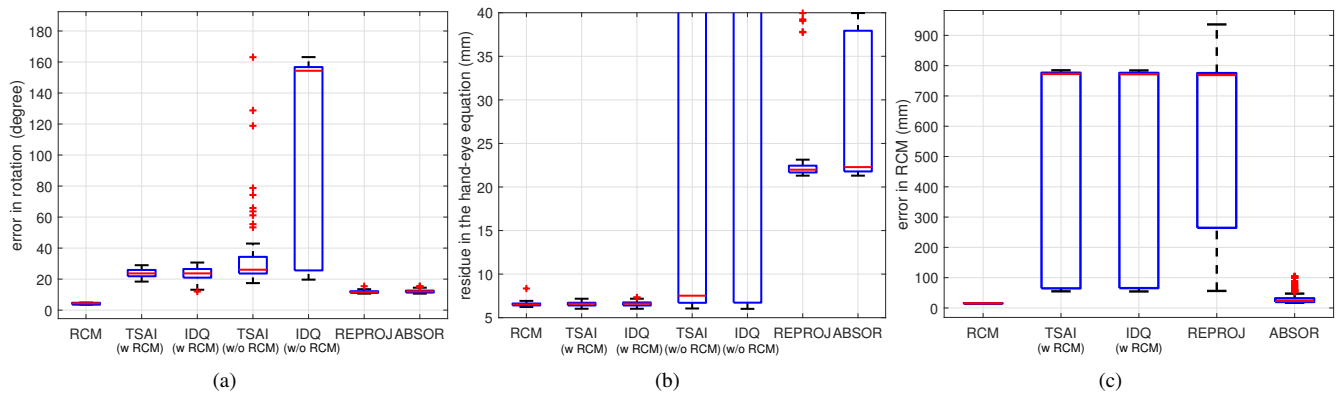
Fig. 10. The comparison between the calibration performances of each algorithm when tested with the real data. The red line represents the median of the distribution and the outliers are marked by a red cross. (a) Error in the rotation estimation. (b) Residue in the hand-eye equation. (c) Error in the RCM re-projection.
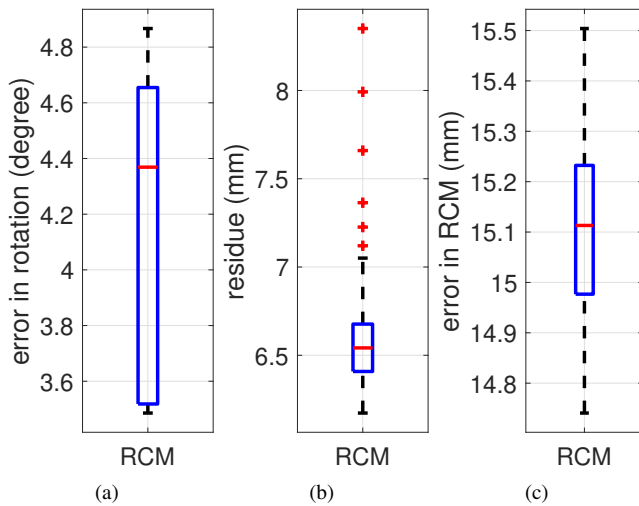

Fig. 11. The calibration performance of the proposed algorithm when tested with the real data. (a) Error in the rotation estimation. (b) Residue in the hand-eye equation. (c) Error in the RCM re-projection.

TSAI, IDQ and REPROJ is not satisfactory. Fig. 10(c) shows that the RCM error is swinging between 25 mm to 770 mm with the median of 750 mm. This indicates that the RCM constrained motions are indeed ill-posed, producing very different transformations with similarly lower equation residues. One of the main reason why TSAI and IDQ fail in the experiment is because the restricted rotational motion in relative motions are extremely small and cannot satisfy the uniqueness criteria in solving the hand-eye problem. This confirms our deduction in Eq. 2 that it cannot be used to calibrate the hand-eye matrix in this situation.

We also compare the RCM method with the RCM-free setup to show the advantage of introducing the RCM constraint to the problem. Fig. 10(a) and Fig. 10(b) show that with the same experimental setup, the proposed method can outperform the RCM-free methods. The result clearly demonstrates that introducing the RCM constraint creates a simpler version of the hand-eye problem than the original one and can yield a more accurate calibration result, although the restricted motion invalidates the RCM-free formulation.

Fig. 11(a)-11(c) show the calibration performance of the RCM algorithm alone. The median of the error is around 4.38 degrees in rotation, 6.6 mm in the residue and 15.1 mm in the RCM re-projection. The calibration error still exists because of the error in robot and camera motions. According to the proof in [8], the influence of noise to the calibrated hand-eye matrix can be reduced by the factor of θ_{ij} in rotation and $2\sin\frac{\theta_{ij}}{2}$ in translation, where θ_{ij} is the angle of rotation between two measurements. Given that the rotational motion is small, the effect of noise cannot be eliminated completely and is propagated to the estimated hand-eye matrix as shown in the calibration result. However, the result clearly indicates that the world-grid transformation $^{\text{world}}\vec{z}_{\text{grid}}$ can be recovered using the calibrated hand-eye matrix which means that the RCM algorithm can calibrate the hand-eye transformation with a restricted motion range and the result agrees with the experiment and with the simulated data. Although the calibration error is still high and may not yet be applicable to the real systems, this shows that introducing the RCM position is one of the solutions to the hand-eye calibration in RMIS. Plus, enforcing this constraint avoid the requirement of surgical robots to have a non-RCM compliant motion during calibration, which is currently not available in any hospital setting. Free-motion hand-eye techniques would require a re-design of surgical robot mechanisms and result in a more complex workflow and more time consuming calibration.

VI. CONCLUSION

We have developed a new hand-eye calibration algorithm incorporating RCM position constraints to enhance the calibration accuracy and practical implementation in such configurations. The algorithm assumes that the z-axis of the camera poses intersect each other at the RCM, thus defining the RCM, and uses a geometrical solution to find the RCM position. This allows us to construct an absolute orientation problem with an additional constraint to the conventional general hand-eye equations. Our method is well-suited to robotic systems that are mechanically constrained around the RCM, such as current keyhole surgery robotic tele-manipulators. In numerical experiments on simulated data, our algorithm outperforms classic hand-eye approaches and working with real data captured

from a KUKA robot equipped with a NanEye stereo camera, our algorithm yields the lowest error with the sensible hand-eye matrix while the other methods fail to calibrate the transformation. This is due to the motion range limitation of the classic formulation of the hand-eye problem which invalidates the solution of the rotational component of the transformation. We also note that our method even outperforms the RCM-free setup in every evaluation criteria. This clearly shows that the proposed algorithm is suitable for applications where practical considerations limit the calibration process and do not permit collection of calibration datapoints with a wide motion range. Future directions of research, in addition to the minimal solver contribution of the RCM constraint, include investigating the feasibility of adding the constraints based on the kinematic structure of the robot to simplify the hand-eye problem even further.

REFERENCES

[1] F. O. Matu, *et al.*, "Stereoscopic augmented reality system for supervised training on minimal invasive surgery robots," in *Proc. 2014 Virt. Real. Int. Conf. (VRIC'2014)*, 2014, pp. 33:1–33:4.

[2] S. Aksungur, "Remote center of motion (rcm) mechanisms for surgical operations," *Int. J. Appl. Math. Electron. Comput.*, vol. 3, no. 2, pp. 119–126, Feb 2015.

[3] G. Dwyer, *et al.*, "A continuum robot and control interface for surgical assist in fetoscopic interventions," *IEEE Robot. Autom. Lett. (RAL)*, vol. 2, no. 3, pp. 1656–1663, July 2017.

[4] D. Stoyanov, "Surgical vision," *Ann. Biomed. Eng.*, vol. 40, no. 2, pp. 332–345, Feb 2012.

[5] N. Aghakhani, *et al.*, "Task control with remote center of motion constraint for minimally invasive robotic surgery," in *2013 IEEE Int. Conf. Robot. Autom. (ICRA'2013)*, May 2013, pp. 5807–5812.

[6] A. Malti, "Handeye calibration with epipolar constraints: Application to endoscopy," *Robot. Autom. Syst.*, vol. 61, no. 2, pp. 161 – 169, 2013.

[7] R. Y. Tsai and R. K. Lenz, "Real time versatile robotics hand/eye calibration using 3d machine vision," in *Proc. 1988 IEEE Int. Conf. Robot. Autom. (ICRA'1988)*, Apr 1988, pp. 554–561 vol.1.

[8] R. Y. Tsai and R. K. Lenz, "A new technique for fully autonomous and efficient 3d robotics hand/eye calibration," *IEEE Tran. Robot. Autom.*, vol. 5, no. 3, pp. 345–358, Jun 1989.

[9] J. Schmidt and H. Niemann, "Data selection for hand-eye calibration: A vector quantization approach," *Int. J. Robot. Res. (IJRR)*, vol. 27, no. 9, pp. 1027–1053, 2008.

[10] K. H. Strobl and G. Hirzinger, "Optimal hand-eye calibration," in *2006 IEEE/RSJ Int. Conf. Intell. Robot. Syst. (IROS'2006)*, Oct 2006, pp. 4647–4653.

[11] F. C. Park and B. J. Martin, "Robot sensor calibration: solving $\mathbf{AX} = \mathbf{XB}$ on the euclidean group," *IEEE Tran. Robot. Autom.*, vol. 10, no. 5, pp. 717–721, Oct 1994.

[12] K. Pachtrachai, *et al.*, "Hand-eye calibration for robotic assisted minimally invasive surgery without a calibration object," in *2016 IEEE/RSJ Int. Conf. Intell. Robot. Syst. (IROS'2016)*, Oct 2016, pp. 2485–2491.

[13] K. Pachtrachai, *et al.*, "Adjoint transformation algorithm for hand–eye calibration with applications in robotic assisted surgery," *Ann. Biomed. Eng.*, Jul 2018.

[14] J. C. K. Chou and M. Kamel, "Quaternions approach to solve the kinematic equation of rotation, $\mathbf{A}_a\mathbf{A}_x = \mathbf{A}_x\mathbf{A}_b$, of a sensor-mounted robotic manipulator," in *Proc. 1988 IEEE Int. Conf. Robot. Autom. (ICRA'1988)*, Apr 1988, pp. 656–662 vol.2.

[15] A. Malti and J. P. Barreto, "Robust hand-eye calibration for computer aided medical endoscopy," in *2010 IEEE Int. Conf. Robot. Autom. (ICRA'2010)*, May 2010, pp. 5543–5549.

[16] K. Daniilidis, "Hand-eye calibration using dual quaternions," *Int. J. Robot. Res. (IJRR)*, vol. 18, no. 3, pp. 286–298, 1999.

[17] J. Heller, D. Henrion, and T. Pajdla, "Hand-eye and robot-world calibration by global polynomial optimization," in *2014 IEEE Int. Conf. Robot. Autom. (ICRA'2014)*, May 2014, pp. 3157–3164.

[18] J. Schmidt, F. Vogt, and H. Niemann, "Calibration–free hand–eye calibration: A structure–from–motion approach," in *Pattern Recognition*. Springer Berlin Heidelberg, 2005, pp. 67–74.

[19] J. Heller, *et al.*, "Structure-from-motion based hand-eye calibration using l_∞ minimization," in *IEEE Conf. Comput. Vis. Pattern Recognit. 2011 (CVPR'2011)*, June 2011, pp. 3497–3503.

[20] M. K. Ackerman, A. Cheng, and G. Chirikjian, "An information-theoretic approach to the correspondence-free $ax = xb$ sensor calibration problem," in *2014 IEEE Int. Conf. Robot. Autom. (ICRA'2014)*, May 2014, pp. 4893–4899.

[21] Q. Ma, H. Li, and G. S. Chirikjian, "New probabilistic approaches to the $ax = xb$ hand-eye calibration without correspondence," in *2016 IEEE Int. Conf. Robot. Autom. (ICRA'2016)*, May 2016, pp. 4365–4371.

[22] F. Schramm, *et al.*, "Calibration free image point path planning simultaneously ensuring visibility and controlling camera path," in *Proc. 2007 IEEE Int. Conf. Robot. Autom. (ICRA'2007)*, April 2007, pp. 2074–2079.

[23] A. Filion, *et al.*, "Robot calibration using a portable photogrammetry system," *Robot. Comput.-Integr. Manuf.*, vol. 49, pp. 77 – 87, 2018.

[24] G. Xiong, *et al.*, "A product-of-exponential-based robot calibration method with optimal measurement configurations," *Int. J. Adv. Robot. Syst. (IJARS)*, vol. 14, no. 6, pp. 1–12, 2017.

[25] H. Zhuang, S. Roth, and R. Sudhakar, "Simultaneous robot/world and tool/flange calibration by solving homogeneous transformation equations of the form ax=yb," *IEEE Tran. Robot. Autom.*, vol. 10, no. 4, pp. 549–554, Aug 1994.

[26] F. Dornaika and R. Horaud, "Simultaneous robot-world and hand-eye calibration," *IEEE Tran. Robot. Autom.*, vol. 14, no. 4, pp. 617–622, Aug 1998.

[27] A. Tabb and K. M. Ahmad Yousef, "Solving the robot-world hand-eye(s) calibration problem with iterative methods," *Mach. Vision. Appl.*, vol. 28, no. 5, pp. 569–590, Aug 2017.

[28] H. Li, *et al.*, "Simultaneous hand-eye and robot-world calibration by solving the $ax = yb$ problem without correspondence," *IEEE Robot. Autom. Lett. (RAL)*, vol. 1, no. 1, pp. 145–152, Jan 2016.

[29] K. Koide and E. Menegatti, "General handeye calibration based on reprojection error minimization," *IEEE Robot. Autom. Lett. (RAL)*, vol. 4, no. 2, pp. 1021–1028, April 2019.

[30] K. Huang and C. Stachniss, "On geometric models and their accuracy for extrinsic sensor calibration," *2018 IEEE Int. Conf. Robot. Autom. (ICRA'2018)*, pp. 1–9, 2018.

[31] Z. Wang, *et al.*, "Vision-based calibration of dual rcm-based robot arms in human-robot collaborative minimally invasive surgery," *IEEE Robot. Autom. Lett. (RAL)*, vol. 3, no. 2, pp. 672–679, April 2018.

[32] F. Vasconcelos, *et al.*, "Relative pose estimation from image correspondences under a remote center of motion constraint," *IEEE Robot. Autom. Lett. (RAL)*, vol. 3, no. 3, pp. 2654–2661, July 2018.

[33] B. K. Horn, "Closed-form solution of absolute orientation using unit quaternions," *J. Opt. Soc. Am. A.*, vol. 4, no. 4, pp. 629–642, 1987.

[34] S. H. Mark W. Spong and M.Vidyasagar, *Robot modeling and control*. John Wiley & Sons, Ltd., 2006.

[35] *ISI API User Guide: da Vinci Research Kit*, Intuitive Surgical Inc., 2010.