

Vision-Based Calibration of Dual RCM-Based Robot Arms in Human-Robot Collaborative Minimally Invasive Surgery

Zerui Wang¹ *Student Member, IEEE*, Ziwei Liu², Qianli Ma³, Alexis Cheng⁴, Yun-hui Liu¹ *Fellow, IEEE*, Sungmin Kim⁴, Anton Deguet⁴, Austin Reiter⁴, Peter Kazanzides⁴ *Member, IEEE*, and Russell H. Taylor⁴ *Life Fellow, IEEE*

Abstract—This paper reports the development of a vision-based calibration method for dual remote center-of-motion (RCM) based robot arms in a human-robot collaborative minimally invasive surgery (MIS) scenario. The method does not require any external tracking sensors and directly uses images captured by the endoscopic camera and the robot encoder readings as calibration data, which leads to a minimal and practical system in the operating room. By taking advantage of the motion constraints imposed by the RCM-based kinematics of the robotic surgical tools and cameras, we can find unique relationships between the endoscope and the surgical tool using camera perspective projection geometry without the geometric information of the tool. A customized vision-based centerline detection algorithm is also proposed, which provides robust estimation of centerline positions for a variety of settings. We validate the method through simulations and an experimental study in a simulated MIS scenario, in which the first generation da Vinci Surgical System controlled by the open source da Vinci Research Kit (DVRK) electronics and cisst/SAW software environment is used.

Index Terms—Calibration and identification, computer vision for medical robotics, surgical robotics: laparoscopy.

I. INTRODUCTION

Nowadays, minimally invasive surgery (MIS) techniques have been applied to many surgical fields (e.g., laparoscopy, arthroscopy, and gynecology) due to their advantages over conventional open surgery [1]. MIS reduces trauma to the body and exposure of the internal organs, and prevents unnecessary blood loss, shortening recovery time, and costing

Manuscript received: February, 15, 2017; Revised May, 21, 2017; Accepted July, 10, 2017.

This work is supported in part by HK RGC under grants 415011 and CUHK6/CRF/13G, by HK ITF under grants ITS/12/15FP and ITT/012/15GP, by VC's Discretionary Fund through CURI grant 4930763 and 4930725, by CUHK SHIAE under grant 8115053, from the NSF under NRI grant 1637789, and by Johns Hopkins University Internal funds.

¹Z. Wang, Y.-H. Liu are with the Department of Mechanical and Automation Engineering, The Chinese University of Hong Kong, HKSAR, China. zrwang, yhliu@mae.cuhk.edu.hk

²Z. Liu is with the Department of Information Engineering, The Chinese University of Hong Kong, HKSAR, China. lz013@ie.cuhk.edu.hk

³Q. Ma is with the Department of Mechanical Engineering, Johns Hopkins University, Baltimore, Maryland, USA. mqianli@jhu.edu

⁴A. Cheng, S. Kim, A. Deguet, A. Reiter, P. Kazanzides and R. H. Taylor are with the Department of Computer Science, Johns Hopkins University, Baltimore, Maryland, USA. acheng22, sungminkim, anton.deguet, areiter2, pkaz, rht@jhu.edu

Digital Object Identifier (DOI): see top of this page.

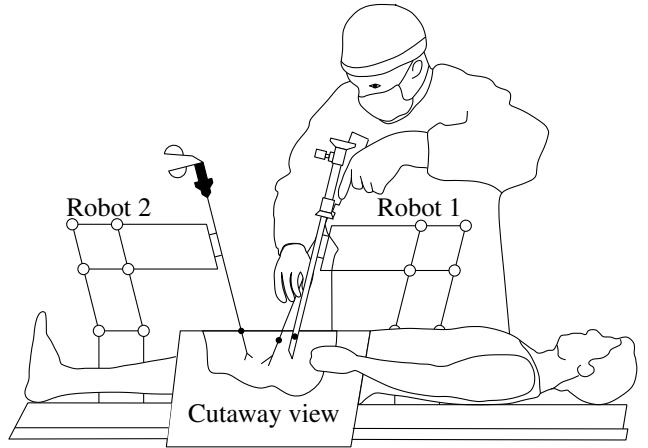


Fig. 1: A human-robot collaborative MIS scenario. Two robots are assisting a surgeon during surgical procedures. One robot is holding an endoscope, and another robot is holding a surgical tool.

less for hospitals [2]. However the technical complexity for surgeons is increased due to the introduction of special surgical tools (endoscopes and long-shafted instruments). Advances from robotics (e.g., motorized position control, stereo 3D imaging systems, and real-time force sensory feedback) have been applied to improve surgeon performance and to remove limitations in MIS surgical environments [3].

In a typical telesurgical system such as the da Vinci surgical system [4], two or more robotic arms manipulate surgical instruments while another arm manipulates a (stereo) endoscopic camera. The robot controller uses kinematic equations of the robot arms to compute the pose of the instrument arms relative to the endoscope's coordinate systems. Although this approach is sufficient for conventional human-in-the loop telesurgery, structural deflections make it problematic for many computer-assisted modes requiring more accurate co-registration of robot arms and the vision system. Similarly, other telesurgical systems may not have a known mechanical connection between endoscope and manipulation arms. In these cases, a further calibration procedure is needed [5].

The hand-eye calibration problem ($\mathbf{AX} = \mathbf{XB}$) and its extension, the robot-world calibration problem ($\mathbf{AX} = \mathbf{YB}$), have been well studied by many researchers. Shiu [6], [7],

Chou [8], [9], and Tsai [10], [11] were among the first to solve the calibration problem. Chen [12], Wang [13], Park [14], Horaud [15], Daniilidis [16], Angeles [17], Andreff [18], Schmidt [19], Zhao [20], [21], Strobl [22], Malti [23], Heller [24], Ruland [25] and Kukelova [26] proposed their methods to solve the $\mathbf{AX} = \mathbf{XB}$ problem. Quite a few $\mathbf{AX} = \mathbf{YB}$ solvers have been proposed by Zhuang [27], Hirsh [28], Li [29], Hu [30], Ernst [31], Shah [32], Pardeep [33], Ha [34] and Ma [35]. The literature mentioned above are all about general robot manipulators in industrial settings. However, the approaches are not suitable for the human-robot collaborative MIS scenario (as shown in Fig. 1) because they all require the use of a calibration grid or a checkerboard for robot-world/robot-robot calibration. Furthermore, some of them require external tracking sensors in addition to the robot [36]. These calibration setups can add a lot of complexity to the surgical environment. Also, the confined workspace within the patient's body may make it infeasible to use a calibration grid or a checkerboard. The closest works to ours are [37] and [38]. Agustinos et al. briefly described the calibration of a transformation between the reference frames of two robotic holders. The key information for the calibration in [37] is the 3D coordinates of the instrument's tip, which is computed by the method proposed in [38]. The method uses manual selection/initialization of instrument centerlines in a sequence of images with one camera position and a moving instrument, which relies on the geometric model of the instrument (e.g., tool diameter).

In this paper, we demonstrate a method for calibrating two remote center-of-motion (RCM) based robot arms¹ (\mathbf{Y} in Fig. 2) in the MIS scenario shown Fig. 1, where one arm is holding an endoscope (monocular camera) and the other one is holding a surgical tool. With a manual offline camera calibration (by [39]), our method does not require any external tracking sensors and directly uses images captured by the endoscopic camera and the robot encoder readings as calibration data. By taking advantage of the motion constraints imposed by the RCM-based kinematics of the robotic surgical tools and cameras, we can find unique relationships between the endoscope and the surgical tool using camera perspective projection geometry without the geometric information of the tool (e.g., tool diameter). Vision-based centerline detection forms a basis for the above calibration procedure. We refer readers to [40] for a comprehensive review. However, existing works on centerline detection either require manual tip selection [41] or impose strong assumptions on the scene layout [42], which are not applicable in our case. Therefore, we briefly presented a customized centerline detection algorithm especially for our calibration method. All these techniques contribute to a robust robot arms calibration system which has demonstrated its effectiveness on both numerical simulations and simulated MIS scenarios.

The rest of this manuscript is organized as follows: In Section II, we present the mathematical models. Section III details the calibration algorithm and the optimization method.

¹The method can be applied to any RCM-based robots (mechanically or by software constraints), e.g., ViKY surgical system, da Vinci surgical system, and KUKA robot manipulator with soft RCM kinematic constraints.

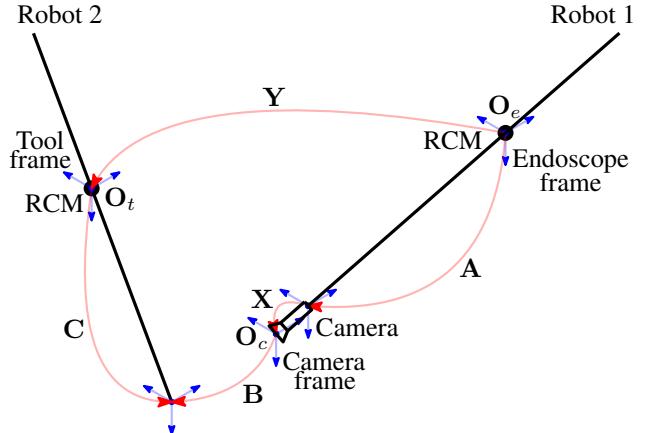


Fig. 2: Illustration shows transformations among different frames of a dual RCM-based robot arm system, where \mathbf{A} and \mathbf{C} are the homogeneous transformations calculated from encoder readings, \mathbf{B} denotes the transformations measured by the camera, \mathbf{X} represents the transformation from the robot end-effector to the mounted camera, and \mathbf{Y} describes the unknown transformation between the base frames of the two robot arms. It also shows three coordinate systems, i.e., tool frame, endoscope frame, and camera frame.

Simulation and its results are shown in Section IV. In Section V, we present the conducted experiments and evaluate the results. Limitations and future improvements are discussed in Section VI.

II. MODELING

A. Notation

In general, we denote a vector and a matrix by a lower and a upper case bold letter, respectively, e.g., $\mathbf{v} \in \mathbb{R}^n$ and $\mathbf{M} \in \mathbb{R}^{p \times q}$. An exception to this notation is the use of the letter “p”. We use bold uppercase \mathbf{P} and lowercase \mathbf{p} to represent points in the Cartesian space and their projections on the image plane, respectively. Quaternions are denoted by an italic letter with a circle above it, e.g., $\hat{\mathbf{q}} = q_0 + q_1\mathbf{i} + q_2\mathbf{j} + q_3\mathbf{k} = (q_0, \mathbf{q})$. Leading superscripts identify which coordinate system a quantity is written in, e.g., ${}^A\mathbf{P}$ represents a position vector described in $\{A\}$. A quantity also possessing a leading subscript specifies a relationship between two coordinate systems, e.g., ${}^A_B\mathbf{R}$ and ${}^A_B\mathbf{T}$ are respectively rotation and homogeneous transformation matrices from $\{A\}$ to $\{B\}$ [43].

B. Human-robot Collaborative MIS Scenario

Consider the human-robot collaborative MIS scenario shown in Fig. 1. In this scenario, a surgeon performs surgical procedures with help from a robot holding an endoscope (Robot 1) and a robot holding a surgical tool (Robot 2). The positions and orientations of the endoscope and the surgical tool are controlled by Robot 1 and Robot 2, respectively. For Robot 1 or Robot 2, its RCM is located at an entry point on the patient's body. At each RCM point, we attach a coordinate frame to it and use it as the base frame to describe the kinematics of the corresponding robot. As shown

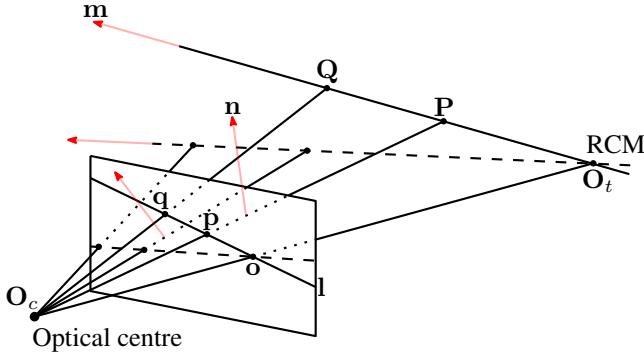


Fig. 3: Projection geometry of the surgical tool on the image plan.

in Fig. 2, the *endoscope frame* ($\{e\}$) represents the frame located at the RCM of Robot 1, and the *tool frame* ($\{t\}$) denotes the frame located at the RCM of Robot 2. Notably, there is another frame named the *camera frame* ($\{c\}$), which is different from the *endoscope frame*, located at the optical center of the endoscopic camera.

C. Perspective Projection Geometry of RCM-Based Robots

A perspective projection model is used for the endoscopic camera. According to the perspective projection geometry, the axis of the surgical tool (m) in the Cartesian space, the projection of this axis into the image plane (l), and the optical center (O_c) of the camera are coplanar (as shown in Fig. 3). Denote any two different points (observable by the camera) on the tool axis by P and Q and their projections by p and q . Then, the normal of the plane defined by O_c , P and Q can be calculated as follows

$${}^c\mathbf{n}(\theta_1, \theta_2) = \begin{bmatrix} \mathbf{p} \\ f \end{bmatrix} \times \mathbf{l} = \begin{bmatrix} \mathbf{p} \\ f \end{bmatrix} \times \begin{bmatrix} \mathbf{q} \\ f \end{bmatrix} \quad (1)$$

where θ_1 and θ_2 represent the joint positions of Robot 1 and Robot 2, respectively, and f is the focal length of the camera. The normal is expressed with respect to the *camera frame*.

Geometrically, the origin (O_t) of the *tool frame* is always in the plane mentioned above, and the axis (m) of the surgical tool is always perpendicular to the normal of the plane. These two geometric relationships can be mathematically modeled by the following equations

$$\begin{bmatrix} {}^c\mathbf{n}(\theta_1, \theta_2) \\ 1 \end{bmatrix}^\top {}^c\mathbf{T}^{-1}(\theta_1) \begin{bmatrix} {}^e\mathbf{O}_t \\ 1 \end{bmatrix} = 1 \quad (2)$$

$${}^c\mathbf{n}^\top(\theta_1, \theta_2) {}^e\mathbf{R}^\top(\theta_1) {}^e\mathbf{R}^t \mathbf{m}(\theta_2) = 0 \quad (3)$$

where ${}^e\mathbf{R}(\theta_1)$ and ${}^c\mathbf{T}(\theta_1)$ represent the rotation and homogeneous transformations from the *endoscope frame* to the *camera frame*, respectively, ${}^e\mathbf{O}_t$ and ${}^e\mathbf{R}$ denotes the translation and rotation from the *endoscope frame* to the *tool frame*, respectively, and ${}^t\mathbf{m}(\theta_2)$ represents the direction of the surgical tool axis expressed in the *tool frame*. ${}^e\mathbf{O}_t$ and ${}^e\mathbf{R}$ are the two quantities that the calibration algorithm is aiming to obtain.

Eq. (2) can be rewritten as follows

$${}^c\mathbf{n}^\top(\theta_1, \theta_2) {}^e\mathbf{R}^\top(\theta_1) {}^e\mathbf{O}_t = {}^c\mathbf{n}^\top(\theta_1, \theta_2) {}^e\mathbf{R}^\top(\theta_1) {}^e\mathbf{O}_c(\theta_1) \quad (4)$$

where ${}^e\mathbf{O}_c(\theta_1)$ denotes the translation from the *endoscope frame* to the *camera frame*.

III. CALIBRATION ALGORITHM

A. Calibrating the Translation Vector

The objective of the calibration is to obtain the homogeneous transformation that consists of the translation ${}^e\mathbf{O}_t$ and the rotation ${}^e\mathbf{R}$. Due to the singularity, the depth of the RCM of Robot 2 cannot be recovered from the images captured by the monocular camera in a fixed view. In order to calculate the translation ${}^e\mathbf{O}_t$, the endoscope must be moved to at least two views, with the motion being non-collinear to the vector defined by the RCM of Robot 2 and the endoscopic optical center. Therefore, by taking a large number of images, captured from at least two views, of the surgical tool in different orientations, it is possible to calculate ${}^e\mathbf{O}_t$ by solving the following optimization problem based on Eq. (4).

$$\arg \min_{{}^e\mathbf{O}_t \in \mathbb{R}^3} \sum_{\theta_1} \sum_{\theta_2} \left\| {}^c\mathbf{n}^\top(\theta_1, \theta_2) {}^e\mathbf{R}^\top(\theta_1) {}^e\mathbf{O}_t - {}^c\mathbf{n}^\top(\theta_1, \theta_2) {}^e\mathbf{R}^\top(\theta_1) {}^e\mathbf{O}_c(\theta_1) \right\|_2^2 \quad (5)$$

B. Calibrating the Rotation Matrix

To calibrate the rotation ${}^e\mathbf{R}$, at least three orientations of the surgical tool should be recorded. Note that ${}^t\mathbf{m}(\theta_2)$ can be calculated from encoder readings of Robot 2. For at least three orientations of the surgical tool, a number of images are captured by the camera in different configurations, then the rotation can be obtained by solving the following optimization problem based on Eq. (3)

$$\arg \min_{{}^e\mathbf{R} \in \text{SO}(3)} \sum_{\theta_1} \sum_{\theta_2} \left\| {}^c\mathbf{n}^\top(\theta_1, \theta_2) {}^e\mathbf{R}^\top(\theta_1) {}^e\mathbf{R}^t \mathbf{m}(\theta_2) \right\|_2^2 \quad (6)$$

where $\text{SO}(3)$ represents special orthogonal group. We present both a linear and a non-linear method for solving this optimization problem.

1) *Linear Optimization*: In Eq. (7), the linear system derived from Eq. (3) can be obtained by introducing the Kronecker Product

$$[({}^t\mathbf{m}^\top(\theta_2)) \otimes ({}^c\mathbf{n}^\top(\theta_2) {}^e\mathbf{R}^\top(\theta_1))] \text{Vec}({}^e\mathbf{R}) = 0 \quad (7)$$

where the definitions of \otimes and Vec -operator can be found in Appendix A.

By solving the following optimization problem, an invertible matrix whose columns are orthogonal to each other can be obtained

$$\arg \min_{\text{Vec}(\mathbf{V}) \in \mathbb{R}^9} \sum_{\theta_1} \sum_{\theta_2} \left\| \left[\left({}^t \mathbf{m}^\top(\theta_2) \right) \otimes \left({}^c \mathbf{n}^\top(\theta_2, \theta_2) {}^c \mathbf{R}^\top(\theta_1) \right) \right] \cdot \text{Vec}(\mathbf{V}) \right\|_2^2 \quad (8)$$

The rotation matrix ${}^e \mathbf{R}$ can be obtained by normalizing the matrix \mathbf{V} [44]

$${}^e \mathbf{R} = \frac{\text{sign}(\det(\mathbf{V}))}{\|\det(\mathbf{V})\|^{\frac{1}{3}}} \mathbf{V} \quad (9)$$

Remark 1. In practice, columns of \mathbf{V} may not be orthogonal to each other due to noise. Before normalization, matrix \mathbf{V} should be projected into an orthogonal matrix using QR decomposition or singular value decomposition (SVD).

2) *Non-linear Optimization:* By introducing a unit quaternion \dot{q} corresponding to the rotation matrix ${}^e \mathbf{R}$, we can rewrite Eq. (6) as follows

$$\begin{aligned} & \arg \min_{{}^e \mathbf{R} \in \text{SO}(3)} \sum_{\theta_1} \sum_{\theta_2} \left\| {}^c \mathbf{n}^\top(\theta_1, \theta_2) {}^e \mathbf{R}^\top(\theta_1) {}^e \mathbf{R} {}^t \mathbf{m}(\theta_2) \right\|_2^2 \\ &= \arg \min_{{}^e \mathbf{R} \in \text{SO}(3)} \sum_{\theta_1} \sum_{\theta_2} \left\| ({}^e \mathbf{R}(\theta_1) {}^c \mathbf{n}(\theta_1, \theta_2)) \cdot ({}^e \mathbf{R} {}^t \mathbf{m}(\theta_2)) \right\|_2^2 \\ &= \arg \min_{\dot{q} \in \text{SU}(2)} \sum_{\theta_1} \sum_{\theta_2} \left\| {}^e \dot{n} \cdot (\dot{q} {}^t \dot{m} \dot{q}^*) \right\|_2^2 \left\| \dot{q} \right\|_2^2 \\ &= \arg \min_{\dot{q} \in \text{SU}(2)} \sum_{\theta_1} \sum_{\theta_2} \left\| ({}^e \dot{n} \dot{q}) \cdot (\dot{q} {}^t \dot{m}) \right\|_2^2 \\ &= \arg \min_{\dot{q} \in \text{SU}(2)} \sum_{\theta_1} \sum_{\theta_2} \left\| \dot{q}^\top \mathbf{N}^\top \mathbf{M} \dot{q} \right\|_2^2 \\ &= \arg \min_{\dot{q} \in \text{SU}(2)} \sum_{\theta_1} \sum_{\theta_2} \left\| \dot{q}^\top \mathbf{W} \dot{q} \right\|_2^2 \end{aligned} \quad (10)$$

where $\text{SU}(2)$ represents special unitary group, i.e., the group of unit quaternions, \dot{q}^* denotes the conjugate of \dot{q} , ${}^c \mathbf{n}(\theta_1, \theta_2) = {}^e \mathbf{R}(\theta_1) {}^c \mathbf{n}(\theta_1, \theta_2)$, ${}^e \dot{n} = (0, {}^e \mathbf{n}(\theta_1, \theta_2))$, and ${}^t \dot{m} = (0, {}^t \mathbf{m}(\theta_2))$. The definition of \mathbf{N} and \mathbf{M} can be found in Appendix B.

C. Calibration Algorithm

Ideally, ${}^e \mathbf{O}_t$ and ${}^e \mathbf{R}$ can be recovered with 6 configurations from 2 poses of Robot 1 and 3 poses of Robot 2.

The novelty of the proposed calibration algorithm lies in the following points:

- 1) Instead of using a checkerboard for recovering the homogeneous transformations (\mathbf{B} in Fig. 2) between the camera attached on one robot and the other robot in a dual robot setup, the proposed algorithm takes advantage of special projection geometry (as shown in Fig. 3) coming from a RCM-based mechanical structure.
- 2) The calibration can be carried out using only laparoscopic images without using any additional external sensors and the instrument geometric models.

Algorithm 1 Calibration of ${}^e \mathbf{O}_t$ and ${}^e \mathbf{R}$

- 1: Find a set of configurations with N_e poses of Robot 1 and N_t poses of Robot 2, provided that for each pose of Robot 1, the instrument of Robot 2 is in the field of view of the camera for all poses of Robot 2.
 - 2: **for** each pose of the Robot 1 **do**
 - 3: **for** each pose of the Robot 2 **do**
 - 4: Record the pairwise data consisting of joint positions of Robot 1 and Robot 2 and images captured by the camera.
 - 5: **end for**
 - 6: **end for**
 - 7: Estimate the position vector ${}^e \mathbf{O}_t$ of the RCM of Robot 2 by solving Eq. (5).
 - 8: Estimate the rotation matrix ${}^e \widehat{\mathbf{R}}$ by solving Eq. (8).
 - 9: Estimate the rotation matrix ${}^e \mathbf{R}$ by solving Eq. (10) with the initial guess $\widehat{q} + \mathcal{N}(0, 1)$.
-

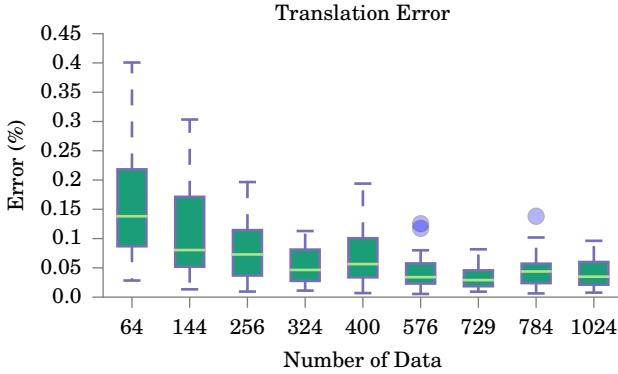
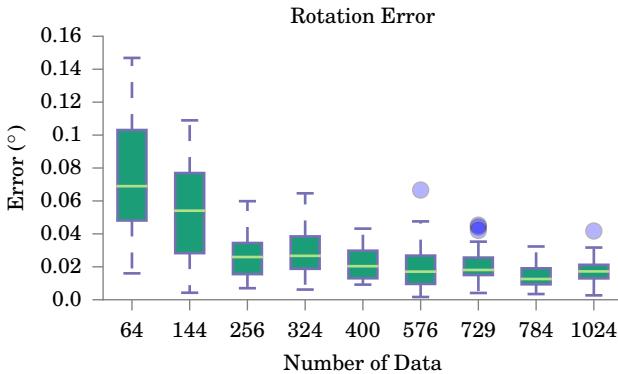
Assumption 1. We assume that the camera intrinsic parameters are known, and the hand-eye transformation from the robot end-effector to the camera is also known (i.e., ${}^e \mathbf{T}$ in ${}^e \mathbf{T}(\theta_1) = {}^e \mathbf{f} \mathbf{T}(\theta_1) {}^e \mathbf{T}$ is known). This assumption does not weaken the feasibility of the algorithm, because the proposed calibration algorithm aims to solve the transformation between the base frames of two robotic arms in the preoperative planning stage [3] of the robot-assisted MIS scenario, in which the arms are first positioned into an appropriate configuration, and then the bases will be fixed during the whole surgery. Therefore, according to this procedure, the calibration will be carried out at the beginning of the preoperative planning stage with offline camera intrinsic and hand-eye calibrations (using [39] and [10]) before it.

Remark 2. The pose combination mentioned in the algorithm is important. Different combinations can affect the calibration accuracy. In the simulations and experimental studies, we found that the full permutation of selected robots' joint positions provides the best results.

Remark 3. The optimization is simplified by using unit quaternions to represent rotations. Converting quaternion fundamental multiplication into matrix multiplication provides faster computations.

Remark 4. Solving rotation by linearization using the Kronecker product may not be as accurate as the non-linear optimization method because projecting the result of the linear optimization method into an orthogonal matrix may lose some accuracy. However, we can use the output (result) of the linear optimization as the input (initial guess) to the non-linear optimization method.

Remark 5. $\mathcal{N}(0, 1)$ in the algorithm represents the standard normal distribution. The initial guess does not necessarily have to be a unit quaternion. An alternative to using the output of the linear optimization method as the initial guess is using a unit quaternion randomly generated from a uniform distribution.

Fig. 4: e_t under different number of configurations.Fig. 5: e_R under different number of configurations.

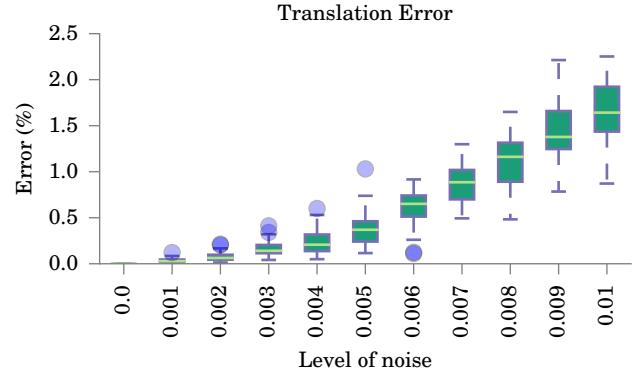
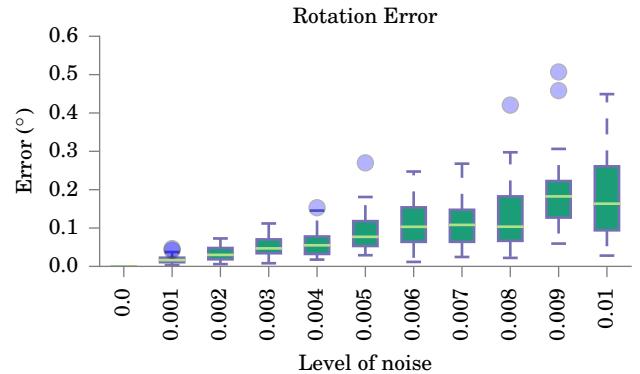
IV. SIMULATION

In practice, inaccurate centerline extraction, hand-eye calibration, and robot forward kinematics are the three major error sources. Therefore, it is necessary to study how these uncertainties affect the calibration result. We conducted numerical simulations to evaluate the overall performance under varying levels of noise, numbers of robot motions, as well as different motion combinations.

A. Simulation Components

The simulations were conducted using MATLAB (The MathWorks Inc.) with Robotics Toolbox [45] and Rviz (3D visualizer).

We used the kinematics models of the Endoscope Camera Manipulator (ECM) and the Patient Side Manipulator (PSM) of the first generation da Vinci Surgical System (Intuitive Surgical Inc.) to simulate Robot 1 and Robot 2, respectively. We built robot kinematics models in both MATLAB and Rviz. The one in MATLAB is for generating synthetic data, and the one in Rviz is for robot model visualization. Each model is able to synchronize its state with the other. This allows one to define a configuration in MATLAB and then visualize it in Rviz, or to select a configuration in Rviz and then update MATLAB configuration to generate synthetic data. Due to robot pose constraints, it is generally more intuitive to generate synthetic data using the robot visualization in Rviz.

Fig. 6: e_t under different levels of noise.Fig. 7: e_R under different levels of noise.

B. Data Generation Process

We first used Rviz to define joint limits, in which the tool attached on the PSM is in the view of the endoscopic camera attached on the ECM. The total poses of the ECM and the PSM are $N_e = n_e^a$ and $N_t = n_t^b$. n_e and n_t are the number of positions for each joint. a and b are the number of joints for each robot. Only the first four joints of the ECM and the first two joints of the PSM contribute to image change, so $a = 4$ and $b = 2$. n_e and n_t are variables in our simulation for varying the number of the configurations.

Reasonable configurations were selected by observing the visualization in Rviz such that the resulted working space is reasonably small for the constraints in the real clinical scenario.

C. Noise Injection

We inject noise to a homogeneous transformation \mathbf{T} in the following manner

$$\mathbf{T}_{\text{noise}} = \mathbf{T} \exp(\delta^\wedge) \quad (11a)$$

$$\delta = \mathcal{N}(\mathbf{0}, \Sigma) \in \mathbb{R}^6 \quad (11b)$$

where $\Sigma = \sigma_{\text{noise}} \mathbb{I}_{6 \times 6}$ represents the covariance matrix that controls the noise level. The hat operator $^\wedge$ converts a 6×1 vector into its corresponding Lie algebra in $se(3)$. Note that Lie algebra element δ is unit-free and one benefit of using the above approach is that the translation and rotation errors can be generated in a cohesive manner.

For each element in the aforementioned pose set, we add noise to the last two frames in the PSM kinematics chain that defines the pose of the surgical tool, and to the last frame in the ECM kinematics chain that defines the pose of the endoscopic camera.

D. Simulation Results

To study the algorithm comprehensively, we conducted the numerical simulations by varying the number of combinations ($N_e \times N_t$) and the noise level σ_{noise} . We ran 30 trials for each set of conditions and drew a boxplot for translation and rotation errors. $e_t = \|\mathbf{t} - \mathbf{t}_{GT}\|/\|\mathbf{t}_{GT}\|$ defines the error of a translation \mathbf{t} , and $e_R = \|\log^\vee(\mathbf{R}^T \mathbf{R}_{GT})\|$ defines the angular error of a rotation \mathbf{R} . \mathbf{t}_{GT} and \mathbf{R}_{GT} represent the translation and rotation ground truth. The $\log(\cdot)$ operator converted an $SE(3)$ matrix into its corresponding element Lie algebra $se(3)$. The inverted hat operator $^\vee$ converts a skew symmetric matrix into its vector form.

First, we fixed the noise level as $\sigma_{\text{noise}} = 0.01$ and varied the number of configurations $N_e \times N_t = \{n_e^4 \times n_t^2 \mid n_e = 2, 3; n_t = 2, 3, 4, 5, 6, 7, 8\}$. As shown in Fig. 4 and 5, as the number of configurations increase, both translation and rotation errors decrease.

Next, we fixed the number of configurations $N_e \times N_t = \{n_e^4 \times n_t^2 \mid n_e = 3; n_t = 3\}$ and applied noise with zero mean and standard deviation σ_{noise} from 0 to 0.01 with a step size of 0.001. As shown in Fig. 6 and 7, average estimation errors of at most 1.6% (2.20 mm) for translation and 0.18° for rotation, are achieved each of the tested noise levels.

We further increased noise level to 0.04 to evaluate which noise level makes the result significantly worse. According to the experiment results, noise level 0.014 makes the translation calibration result much worse (error jumps from 2.207% (3.03 mm) to 3.075% (4.23 mm)), and noise level 0.022 makes the rotation calibration result much worse (error jumps from 0.274° to 0.478°).

For evaluating the robustness of proposed non-linear solver to initial guess, we fixed the noise level to 0.01 and the number of configurations to $N_e \times N_t = \{n_e^4 \times n_t^2 \mid n_e = 3; n_t = 3\}$, and ran the solver for 100 times with randomly generated initial guesses (using uniformly distributed random unit quaternions). 75 trials give the best estimations ($e_R = 0.009^\circ$), and 90 trials give good enough estimations ($\max\{e_R\} = 0.117^\circ$).

V. EXPERIMENTS

A. Experimental Setup

Da Vinci Research Kit (DVRK) [46] was used as the experimental platform. The research kit consists of a first generation da Vinci Surgical System, which provides three PSMs and one ECM on the patient side and two Master Tool Manipulators (MTMs) for teleoperation on the master side, open source electronics, and cisst/SAW software environment.

Before the calibration, we carried out camera intrinsic parameter calibration and hand-eye calibration using the method in [39] and [10], respectively.

During the experiment, we used an ECM holding an endoscope as Robot 1 and a PSM holding a surgical tool (e.g.,

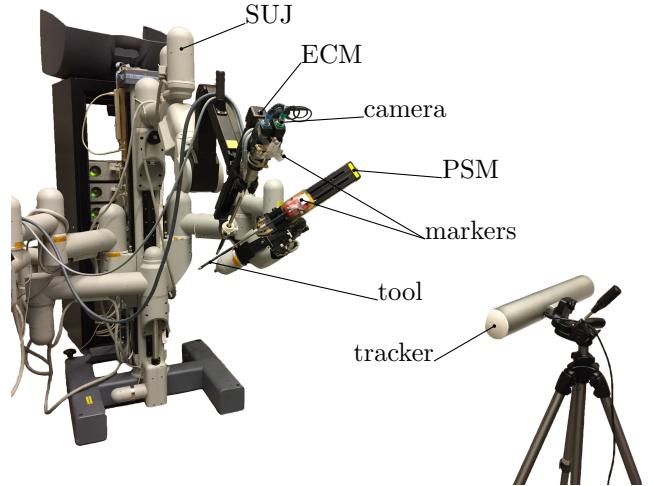


Fig. 8: Experimental platform.

large needle driver) as Robot 2 (as shown in Fig. 8). We fixed the setup joints (i.e., passive joints for holding the active arms) and adjusted the poses of the PSM and the ECM so that the surgical tool is always in the endoscopic field of view. We only used the left channel of the stereo endoscope to simulate a mono endoscopic camera.

In order to obtain the ground truth, we used a commercially available optical tracking system (fusionTrack 500 from Atracsys). We attached passive markers to both PSM and ECM. First we obtained the position of RCMs via pivot calibration. Then, we moved the first joint of the robots independently while fixing the other joint angles to zero. We repeat this for the second joint of the robots to obtain the relative orientation between the two robot base frames.

B. Automatic Centerline Detection

Vision-based centerline detection provides the position and orientation parameters of motorized instruments for the subsequent calibration tasks. It is a crucial step towards fully-automated control of robot-assisted surgery, especially under the circumstances where traditional checkerboard calibration is no longer available. Here, we briefly present a customized centerline detection algorithm using the data recorded during the cadaver or animal experiments. Specifically, our centerline detection algorithm exploits the geometric structure of motorized instrument by directly parameterizing its line shapes. Furthermore, robust estimation and outlier rejection mechanism are incorporated to make our algorithm resilient to background clutters.

1) Local-evidence-based Centerline Detection: We formulate centerline detection as a single-image line fitting problem following [47], which produces a set of line segments as well as their strengths within a given image. Local evidences (e.g. image gradients) are used to initialize and localize the potential edge regions. The gradient threshold is set to 5.22 such that non-edge angle tolerance is 22.5° [47]. Then we adopt the combination of hill climbing search and least square fitting to determine the actual line segments by finding connected edge pixels. Proper pre-processing (e.g. low-pass filtering with kernel size of 11×11) and post-processing (e.g. false positive

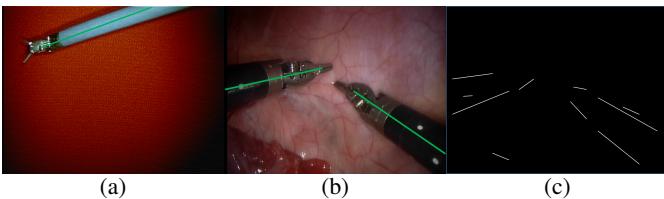


Fig. 9: Experimental results of centerline detection.

rejection with line fitting error threshold of 0.125) techniques are also employed to alleviate the influence of changing lighting conditions, and thus improve the robustness of the whole system.

2) Centerline Detection Experimental Results: The experimental results of centerline extraction algorithms were illustrated in Fig. 9. We can see that our approach is capable of detecting the surgical tool centerlines accurately on both simulated benchmark (Fig. 9 (a)) and field-collected samples (Fig. 9 (b)). We also illustrate the intermediate line segments generated by our algorithm in Fig. 9 (c).

C. Calibration Experimental Results

In the experiment, we chose $n_e = 2$ and $n_t = 4$ such that $N_e \times N_t = 256$. We repeated the experiment with the same configuration 5 times. The experimental results are shown in Table I. And part of the experiment snapshots of the centerline reprojection are shown in Fig. 10.

The errors mainly come from the following aspects: inaccurate camera intrinsic calibration, robot forward kinematics, hand-eye calibration, and centerline extraction. The cable-driven nature of the robots used in the experiment makes absolute quantities calculated from the robot forward kinematics more inaccurate. In addition, errors in the camera intrinsic calibration and centerline extraction also contribute to the inaccuracy of the calibration results. Further, simply using the middle line of the instrument edges as the centerline is an approximation due to the perspective projection. Moreover, the measured ground truth may not be accurate either. All of them mentioned above contribute to the total error in the experimental results.

For a better evaluation of the calibration results, the expected accuracy is defined for applications of teleoperation using da Vinci Classic Surgical System. The transformation from the camera frame of the endoscope mounted on ECM to the base frame of PSM is the key information for teleoperation, which is calculated from the forward kinematics of ECM, PSM and their setup joints. According to the previous work [48], the accuracy of the setup joints of da Vinci S Surgical System (a well improved, newer generation with better accuracy) is 10.6 ± 22.9 mm. Although, the results of the experiment are worse than those of the simulation, the algorithm performs well in the experiments. Notably, the average translation error (7.84%, 16.08 mm) is smaller than the one (17.30%, 35.48 mm) calculated from the forward kinematics of setup joints, and the one of the newer generation with better accuracy (10.6 ± 22.9 mm) in the literature.

TABLE I: The Experimental Results of Five Trials.

Trial (NO.)	1	2	3	4	5	avg.
e_t (%)	7.19	7.41	7.57	8.53	8.50	7.84
e_R (°)	7.85	5.98	8.44	8.54	7.35	7.63

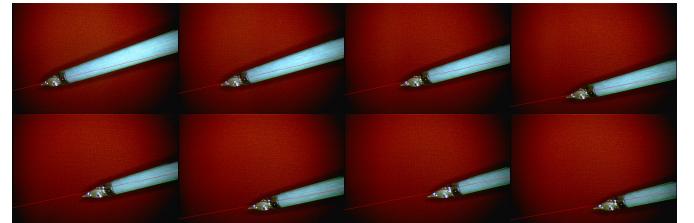


Fig. 10: Snapshots show the reprojection of the centerline.

VI. CONCLUSION

In this paper, we demonstrated a vision-based calibration method for dual remote center-of-motion (RCM) based robot arms in a human-robot collaborative MIS scenario. The method does not require any external tracking sensors and directly uses images captured by the endoscopic camera and the robot encoder readings as calibration data. By taking advantage of the motion constraints imposed by the RCM-based kinematics of the robotic surgical tools and cameras, we could find unique relationships between the endoscope and the surgical tool using camera perspective projection geometry. Domain-specific vision algorithms were also developed to extract the centerline of the surgical tool from camera images. We validated the method through simulations and an experimental study in a simulated MIS scenario, in which the first generation da Vinci Surgical System controlled by the open source da Vinci Research Kit (dVRK) electronics and cisst/SAW software environment is used.

In the future, we are going to study the impact of centerline detection errors quantitatively and use it as a guidance to improve the calibration accuracy regarding centerline detection approach and calibration algorithm. We will also test the improved calibration method in ex vivo experiments.

APPENDIX A

Kronecker Product and Vec-operator are defined as follows (details in [49])

$$\mathbf{A} \otimes \mathbf{B} = \begin{bmatrix} a_{11}\mathbf{B} & \cdots & a_{1q}\mathbf{B} \\ \vdots & & \vdots \\ a_{p1}\mathbf{B} & \cdots & a_{pq}\mathbf{B} \end{bmatrix}$$

$$\text{Vec}(\mathbf{A}) = (a_{11}, \dots, a_{p1}, a_{12}, \dots, a_{p2}, \dots, a_{1q}, \dots, a_{pq})^T$$

APPENDIX B

\mathbf{N} and \mathbf{M} are defined as follows (details in [50])

$${}^e\hat{n}\hat{q} = \begin{bmatrix} 0 & -{}^e n_1 & -{}^e n_2 & -{}^e n_3 \\ {}^e n_1 & 0 & -{}^e n_3 & {}^e n_2 \\ {}^e n_2 & {}^e n_3 & 0 & -{}^e n_1 \\ {}^e n_3 & -{}^e n_2 & {}^e n_1 & 0 \end{bmatrix} \hat{q} = \mathbf{N} \hat{q}$$

$$\ddot{\vec{q}}^t \ddot{\vec{m}} = \begin{bmatrix} 0 & -{}^t m_1 & -{}^t m_2 & -{}^t m_3 \\ {}^t m_1 & 0 & {}^t m_3 & -{}^t m_2 \\ {}^t m_2 & -{}^t m_3 & 0 & {}^t m_1 \\ {}^t m_3 & {}^t m_2 & -{}^t m_1 & 0 \end{bmatrix} \ddot{\vec{q}} = \mathbf{M} \ddot{\vec{q}}$$

REFERENCES

- [1] G. R. Jatzko, P. H. Lisborg, and et al., "Multivariate comparison of complications after laparoscopic cholecystectomy and open cholecystectomy," *Annals of surgery*, vol. 221, no. 4, p. 381, 1995.
- [2] C. C. Laparoscopic and et al., "Laparoscopic surgery versus open surgery for colon cancer: short-term outcomes of a randomised trial," *The lancet oncology*, vol. 6, no. 7, pp. 477–484, 2005.
- [3] R. H. Taylor and D. Stoianovici, "Medical robotics in computer-integrated surgery," *IEEE Trans. Robot. Autom.*, vol. 19, no. 5, pp. 765–781, 2003.
- [4] G. S. Guthart and J. K. Salisbury, "The intuitive/sup tm/telesurgery system: overview and application," in *IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, vol. 1, 2000, pp. 618–621.
- [5] D. Stoyanov, A. Darzi, and G.-Z. Yang, "Laparoscope self-calibration for robotic assisted minimally invasive surgery," in *Int. Conf. Med. Image Comput. Comput. Assist. Interv.* Springer, 2005, pp. 114–121.
- [6] Y. Shiu and S. Ahmad, "Finding the mounting position of a sensor by solving a homogeneous transform equation of the form $ax = xb$," in *IEEE Trans. Robot. Autom.*, vol. 4, 1987, pp. 1666–1671.
- [7] Y. C. Shiu and S. Ahmad, "Calibration of wrist-mounted robotic sensors by solving homogeneous transform equations of the form $ax = xb$," *IEEE Trans. Robot. Autom.*, vol. 5, no. 1, pp. 16–29, 1989.
- [8] J. C. Chou and M. Kamel, "Quaternions approach to solve the kinematic equation of rotation of a sensor-mounted robotic manipulator," in *IEEE Int. Conf. Robot. Autom.*, 1988, pp. 656–662.
- [9] C. Chou and M. Kamel, "Finding the position and orientation of a sensor on a robot manipulator using quaternions," *Int. J. Rob. Res.*, vol. 10, no. 3, pp. 240–254, 1991.
- [10] 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, 1989.
- [11] R. K. Lenz and R. Y. Tsai, "Calibrating a cartesian robot with eye-on-hand configuration independent of eye-to-hand relationship," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 11, no. 9, pp. 916–928, 1989.
- [12] H. H. Chen, "A screw motion approach to uniqueness analysis of head-eye geometry," in *IEEE Conf. Computer Vision and Pattern Recognition*, 1991, pp. 145–151.
- [13] C.-C. Wang, "Extrinsic calibration of a vision sensor mounted on a robot," *IEEE Trans. Robot. Autom.*, vol. 8, no. 2, pp. 161–175, 1992.
- [14] 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, 1994.
- [15] R. Horaud and F. Dornaika, "Hand-eye calibration," *Int. J. Rob. Res.*, vol. 14, no. 3, pp. 195–210, 1995.
- [16] K. Daniilidis, "Hand-eye calibration using dual quaternions," *Int. J. Rob. Res.*, vol. 18, no. 3, pp. 286–298, 1999.
- [17] J. Angeles, G. Soucy, and F. P. Ferrie, "The online solution of the hand-eye problem," *IEEE Trans. Robot. Autom.*, vol. 16, no. 6, pp. 720–731, 2000.
- [18] N. Andreff, R. Horaud, and B. Espiau, "Robot hand-eye calibration using structure-from-motion," *Int. J. Rob. Res.*, vol. 20, no. 3, pp. 228–248, 2001.
- [19] J. Schmidt, F. Vogt, and H. Niemann, "Calibration-free hand-eye calibration: a structure-from-motion approach," in *Joint Pattern Recognition Symposium*. Springer, 2005, pp. 67–74.
- [20] Z. Zhao and Y. Liu, "Hand-eye calibration based on screw motions," in *IEEE Int. Conf. Pattern Recognition*, vol. 3, 2006, pp. 1022–1026.
- [21] Z. Zhao, "Hand-eye calibration using convex optimization," in *IEEE Int. Conf. Robot. Autom.*, 2011, pp. 2947–2952.
- [22] K. H. Strobl and G. Hirzinger, "Optimal hand-eye calibration," in *IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 2006, pp. 4647–4653.
- [23] A. Malti and J. P. Barreto, "Robust hand-eye calibration for computer aided medical endoscopy," in *IEEE Int. Conf. Robot. Autom.*, 2010, pp. 5543–5549.
- [24] J. Heller, M. Havlena, A. Sugimoto, and T. Pajdla, "Structure-from-motion based hand-eye calibration using l minimization," in *IEEE Conf. Computer Vision and Pattern Recognition*, 2011, pp. 3497–3503.
- [25] T. Ruland, T. Pajdla, and L. Krüger, "Globally optimal hand-eye calibration," in *IEEE Conf. Computer Vision and Pattern Recognition*, 2012, pp. 1035–1042.
- [26] Z. Kukelova, J. Heller, and T. Pajdla, "Hand-eye calibration without hand orientation measurement using minimal solution," in *Asian Conf. Computer Vision*. Springer, 2012, pp. 576–589.
- [27] H. Zhuang, Z. S. Roth, and R. Sudhakar, "Simultaneous robot/world and tool/flange calibration by solving homogeneous transformation equations of the form $ax = xb$," *IEEE Trans. Robot. Autom.*, vol. 10, no. 4, pp. 549–554, 1994.
- [28] R. L. Hirsh, G. N. DeSouza, and A. C. Kak, "An iterative approach to the hand-eye and base-world calibration problem," in *IEEE Int. Conf. Robot. Autom.*, vol. 3, 2001, pp. 2171–2176.
- [29] A. Li, L. Wang, and D. Wu, "Simultaneous robot-world and hand-eye calibration using dual-quaternions and kronecker product," *Int. J. Phys. Sci.*, vol. 5, no. 10, pp. 1530–1536, 2010.
- [30] J.-S. Hu and Y.-J. Chang, "Calibration of an eye-to-hand system using a laser pointer on hand and planar constraints," in *IEEE Int. Conf. Robot. Autom.*, 2011, pp. 982–987.
- [31] F. Ernst, L. Richter, L. Matthäus, V. Martens, R. Bruder, A. Schlaefer, and A. Schweikard, "Non-orthogonal tool/flange and robot/world calibration," *Int. J. Med. Rob. Comput. Assisted Surg.*, vol. 8, no. 4, pp. 407–420, 2012.
- [32] M. Shah, "Solving the robot-world/hand-eye calibration problem using the kronecker product," *J. Mech. Rob.*, vol. 5, no. 3, p. 031007, 2013.
- [33] V. Pradeep, K. Konolige, and E. Berger, "Calibrating a multi-arm multi-sensor robot: A bundle adjustment approach," in *Experimental robotics*. Springer, 2014, pp. 211–225.
- [34] J. Ha, D. Kang, and F. C. Park, "A stochastic global optimization algorithm for the two-frame sensor calibration problem," *IEEE Trans. Ind. Electron.*, vol. 63, no. 4, pp. 2434–2446, 2016.
- [35] Q. Ma, Z. Goh, and G. S. Chirikjian, "Probabilistic approaches to the $axb = ycz$ calibration problem in multi-robot systems," in *Proceedings of Robotics: Science and Systems*, Ann Arbor, Michigan, June 2016.
- [36] P. Wilkening, F. Alambeigi, R. J. Murphy, R. H. Taylor, and M. Armand, "Development and experimental evaluation of concurrent control of a robotic arm and continuum manipulator for osteolytic lesion treatment," *IEEE Rob. Autom. Lett.*, vol. 2, no. 3, pp. 1625–1631, 2017.
- [37] A. Agustinos, R. Wolf, J. Long, P. Cinquin, and S. Voros, "Visual servoing of a robotic endoscope holder based on surgical instrument tracking," in *IEEE RAS & EMBS Int. Conf. on Biomedical Robotics and Biomechatronics*, 2014, pp. 13–18.
- [38] R. Wolf, J. Duchateau, P. Cinquin, and S. Voros, "3d tracking of laparoscopic instruments using statistical and geometric modeling," in *Int. Conf. on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2011, pp. 203–210.
- [39] Z. Zhang, "A Flexible New Technique for Camera Calibration (Technical Report)," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 11, pp. 1330–1334, 2002.
- [40] D. Bouget, M. Allan, D. Stoyanov, and P. Jannin, "Vision-based and marker-less surgical tool detection and tracking: a review of the literature," *Medical Image Analysis*, vol. 35, pp. 633–654, 2017.
- [41] S. Voros, J.-A. Long, and P. Cinquin, "Automatic detection of instruments in laparoscopic images: A first step towards high-level command of robotic endoscopic holders," *Int. J. of Rob. Res.*, vol. 26, no. 11–12, pp. 1173–1190, 2007.
- [42] J. Zhou and S. Payandeh, "Visual tracking of laparoscopic instruments," *J. Autom. Cont. Eng. Vol.*, vol. 2, no. 3, pp. 234–241, 2014.
- [43] J. J. Craig, *Introduction to robotics: mechanics and control*. Pearson Prentice Hall Upper Saddle River, 2005, vol. 3.
- [44] N. Andreff, R. Horaud, and B. Espiau, "On-line hand-eye calibration," in *IEEE Int. Conf. 3-D Digi. Imag. Model.*, 1999, pp. 430–436.
- [45] P. I. Corke, "A robotics toolbox for matlab," *IEEE Rob. Autom. Mag.*, vol. 3, no. 1, pp. 24–32, 1996.
- [46] P. Kazanzides, Z. Chen, A. Deguet, G. S. Fischer, R. H. Taylor, and S. P. DiMaio, "An open-source research kit for the da vinci® surgical system," in *IEEE Int. Conf. Robot. Autom.*, 2014, pp. 6434–6439.
- [47] C. Akinlar and C. Topal, "Edlines: A real-time line segment detector with a false detection control," *Pattern Recognit. Lett.*, vol. 32, no. 13, pp. 1633–1642, 2011.
- [48] D. M. Kwartowitz, M. I. Miga, S. D. Herrell, and R. L. Galloway, "Towards image guided robotic surgery: Multi-arm tracking through hybrid localization," *Int. J. Comput. Assist. Radiol. Surg.*, vol. 4, no. 3, pp. 281–286, 2009.
- [49] H. Neudecker, "A note on kronecker matrix products and matrix equation systems," *SIAM J. Appl. Math.*, vol. 17, no. 3, pp. 603–606, 1969.
- [50] B. K. Horn, "Closed-form solution of absolute orientation using unit quaternions," *JOSA A*, vol. 4, no. 4, pp. 629–642, 1987.