2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

Automatic Field of View Adjustment of an RCM Constraint-Free Continuum Laparoscopic Robot

Jing Zhang[1], Baichuan Wang[1], Zhijie Pan[1], Weiqi Li[1] and Mengtang Li[1*]

Abstract— **Automatic laparoscopic field of view (FOV) adjustment can effectively assist surgeons in minimally invasive surgery (MIS). However, existing work based on rod-shaped laparoscopes is inevitably constrained by the remote center of motion (RCM) during the process of FOV adjustment. The RCM limits laparoscopic movement and makes modeling and control more complex. This paper proposes a novel tendon-driven continuum laparoscope that is not affected by the RCM constraint. Furthermore, an automatic FOV adjustment method is designed for the proposed laparoscope robot, which considers the surgical instrument position and size in the image, as well as eye-hand consistency. Two simulation platforms are developed using MATLAB and Webots to intuitively study and optimize the proposed adjustment method. The convergence time of surgical tool tracking with a complex 3D trajectory is only 1s, the average tracking error after stabilization is about 9.97 pixels, and the maximum eye-hand error is only 0.04°. A first-generation prototype is built to verify the tracking performance of the proposed tendon-driven continuum laparoscope. The experimental results show that the proposed system can perform real-time laparoscopic FOV adjustment without being constrained by the RCM.**

I. INTRODUCTION

Minimally invasive surgery (MIS) has become an alternative to traditional open surgery, offering advantages such as less intraoperative bleeding, lower risk of infection, fewer complications, and faster postoperative recovery [1]. During the traditional MIS process, the laparoscope is manually adjusted by an assistant according to the verbal instructions of the chief surgeon. This adjustment method may affect the quality of the surgical field of view (FOV) due to physiological tremors of the assistant's hands or misunderstandings of the chief surgeon's instructions. Therefore, the development of an automated robot-assisted laparoscopic FOV adjustment system is considered a key step in improving the safety and effectiveness of MIS. Traditional methods such as pedals, head and eye movements, and speech commands have been proposed. However, these control methods require extra preoperative training and impose additional burdens on the chief surgeon during the surgical process [2].

To tackle these issues, image-based laparoscopic visual servo control for automatic FOV adjustment has been proposed recently. Osa et al. designed a comprehensive framework for automatic FOV adjustment using visual servoing [3]. The trocar constraint was considered in kinematic modeling, which was recognized as the remote center of motion (RCM) constraint in later work. However, the position of the surgical instrument required additional markers by Osa. The size of the surgical instrument tip in the image and eye-hand consistency were not considered either. To avoid using extra markers, various deep-learning methods have been proposed to achieve marker-free surgical instrument tip localization, such as DSRD-Net [4] and FUN-SIS [5]. Deep learning methods, like CNN-based location method [6] and SAM [7], have been applied in laparoscopic FOV adjustment successfully, which improved the practicality. Zhang et al. considered the central position and size of the surgical instrument tip in the image as well as eye-hand consistency when adjusting FOV [8], which was one of the most comprehensive works currently. A more detailed review on robot-assisted laparoscopic FOV adjustment can be found in [2].

Nevertheless, the traditional rod-shaped rigid laparoscope has disadvantages such as bulk size [9], and limited FOV [10], due to the RCM constraint. Laparoscopes with a flexible tip offered a way to improve flexibility and dexterity. Wang et al. designed a novel robotic laparoscope with a continuous mechanism and conducted experiments to verify its structural performance [11]. Chiu and Li et al. have done a series of excellent works in improving the performance of flexible laparoscopes in surgical instrument tracking [12], [13]. Yet, their laparoscopic adjustment process still relied on an external robotic arm, suffering from the RCM constraint.

As a step forward to tackle the aforementioned obstacles, this paper proposes a tendon-driven continuum laparoscope and its automatic FOV adjustment method. The main contributions of this paper are as follows:

1) A novel 6-degrees-of-freedom (DoF) tendon-driven continuum laparoscope is proposed, which avoids the influence of the RCM constraint during laparoscopic movement.
2) An automatic FOV adjustment framework is presented based on the proposed tendon-driven continuum laparoscope. The adjustment process considers the instrument position and size in the image, and eye-hand consistency.
3) Two simulation platforms are developed under MATLAB and Webots environments for testing and optimizing the proposed automatic FOV adjustment algorithm.

The remainder of this paper is arranged as follows: Section II introduces the mechanical design and kinematic model-

This work was supported by National Natural Science Foundation of China under Grant 6240020272, Foundation for Shenzhen Science and Technology Program under Grant RCBS20221008093104018, and Guangdong Basic and Applied Basic Research Foundation under Grant 2023A1515110440.

Zhang, J., Wang, B., Pan, Z., and Li, W. contributed equally to this work. Li, M. is the corresponding author: limt29@mail.sysu.edu.cn. All the authors are with School of Intelligent Systems Engineering, Shenzhen Campus of Sun Yat-sen University, Shenzhen, China.

ing of the proposed tendon-driven continuum laparoscope. Section III elaborates on the automatic control method of the laparoscope using image-based visual servoing. Section IV presents a detailed simulation study. Experimental verification based on a prototype demonstrating the practical application of the proposed system is presented in Section V. Section VI provides concluding remarks.

II. Design and Kinematic Modeling of Tendon-driven Continuum Laparoscope

This section introduces the mechanical design and corresponding kinematic modeling of the proposed tendon-driven continuum laparoscope, which is fundamental to the subsequent development of the visual servo control method.

A. Mechanical Design

Fig. 1 depicts the structural design of the proposed tendon-driven continuum laparoscope. The entire device generally consists of two modules: the driving module and the continuous laparoscopic module. The stretching of tendons is achieved through the motor screw module, as shown in the inset of Fig. 1. One end of each tendon is fixed to the connector of the driving module, while the other end is fixed to the corresponding disk of the continuum laparoscopic robot. A connector is installed with a flange plate containing a threaded inner hole. When a BLDC motor drives the lead screw to rotate through a shaft coupling, the connector undergoes linear motion rather than rotation. This occurs because the connector is equipped with a linear slider on the other side. Hence, the elongation of a single tendon can be controlled via a BLDC motor.

The continuum laparoscopic module comprises a tendon-driven continuum robot with two universal joints and a translational joint, along with a camera view angle rotational joint. The camera module is positioned at the end of the continuum robot, with its specific mechanism illustrated in the inset of Fig. 1. A tendon passes through a reserved hole in the support chassis, changes direction by bypassing the cylindrical shaft, and wraps around the rotating part. It then passes through another reserved hole on the opposite side by bypassing the other end of the shaft. The camera is securely attached to the rotating part. Consequently, the rotating part can be driven to rotate due to the friction of the tendon when one end of the tendon is shortened and the other end is extended, facilitating camera view angle adjustment.

The laparoscopic robot has a total of 6 DoF and requires 9 driving motors. Among them, 8 motor driving modules are horizontally and symmetrically placed within two layers. This layout can make the entire device more compact. These 8 motor driving modules are used to control the 4-DoF of the continuum robot and the 1-DoF of the rotation of the camera. The stretching direction of each tendon is converted from horizontal to vertical through a V-shaped pulley. The last motor, which is placed vertically, is used to control the translational movement of the entire continuum laparoscope in the vertical direction. The above structural design ensures that the laparoscope only needs to generate motion along

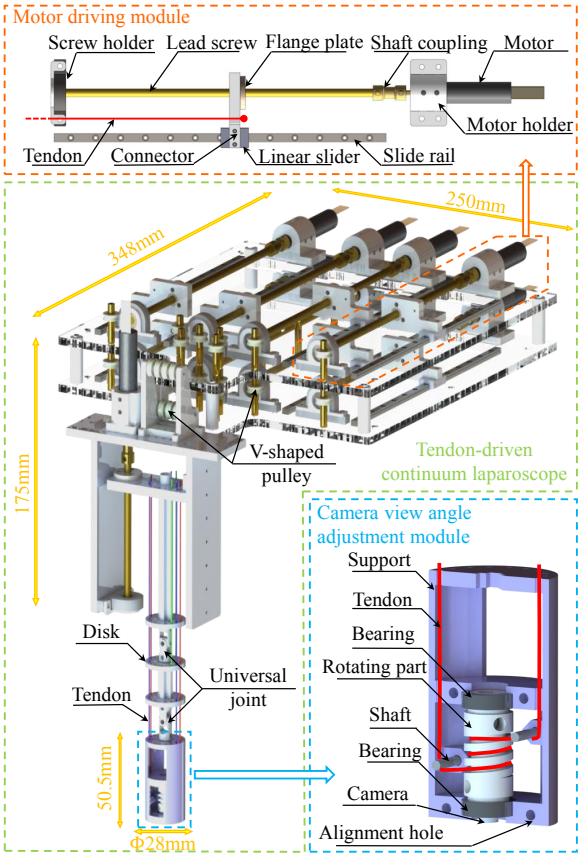

Fig. 1. CAD design of the tendon-driven continuum laparoscope.

the axis of the trocar, where the RCM constraint normally occurs for rod-shaped rigid laparoscopes. Therefore, the proposed 6-DoF laparoscopic robot is not affected by the RCM constraint.

B. Kinematic Modeling of the Continuum Laparoscope

1) Forward Kinematics: As shown in Fig. 2(a), the camera coordinate frame $\{c\}$ is located at the center of the laparoscope's end, which coincides with the center of the camera's end. The initial configuration of the laparoscope's end in the fixed coordinate frame $\{s\}$ is:

$$\boldsymbol{M} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & Z_0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \in SE(3) \tag{1}$$

where Z_0 is the initial height of the laparoscope's end in the frame $\{s\}$.

The tendon-driven continuum laparoscope has 6-DoF, and the corresponding screw axes for each joint are shown in Fig. 2(a). The list of the corresponding screw axes under frame $\{s\}$ is:

$$\boldsymbol{S} = \begin{bmatrix} 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 \\ 0 & -H_1 & 0 & 0 & -H_2 & 0 \\ 0 & 0 & H_1 & H_2 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{2}$$

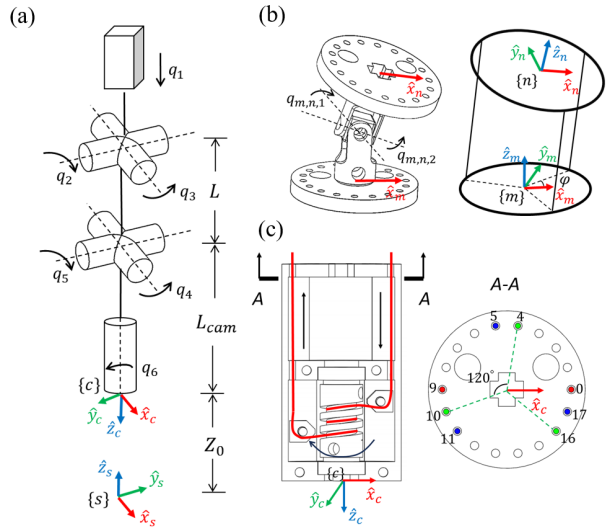

Fig. 2. Schematics of the tendon-driven continuum laparoscope. (a) Initial configuration and joint axes. (b) A single tendon-driven joint. (c) Cross-sectional view of the rotational joint (the camera angle adjustment module). The section view *A-A* displays the distribution of tendons where each color represents one group of two or three tendons.

where $\boldsymbol{S}_i = [\boldsymbol{\omega}_i; \boldsymbol{v}_i] \in \mathbb{R}^{6\times 1}, i = 1,2,\cdots,6$ represents the corresponding screw axis of the i-th joint. $H_1 = L + L_{cam} + Z_0$ and $H_2 = L_{cam} + Z_0$, where L is the distance between the centers of two universal joints, and L_{cam} represents the distance from the center of the camera's end to the center of its adjacent universal joint. The universal joint closer to the camera is denoted by U_2, and the other one is denoted by U_1 in this paper.

The forward kinematics can be therefore calculated by using the Product of Exponentials Formula:

$$\boldsymbol{T}_{sc}(\boldsymbol{\Theta}) = e^{[\boldsymbol{S}_1]q_1} e^{[\boldsymbol{S}_2]q_2} e^{[\boldsymbol{S}_3]q_3} e^{[\boldsymbol{S}_4]q_4} e^{[\boldsymbol{S}_5]q_5} e^{[\boldsymbol{S}_6]q_6} \boldsymbol{M} \in SE(3) \tag{3}$$

where $[\boldsymbol{S}_i] \in se(3)$ is the matrix representation of $\boldsymbol{S}_i$ and $\boldsymbol{\Theta} = [q_1; q_2; q_3; q_4; q_5; q_6]$ is the vectorized joint coordinate.

The end-effector's twist $\boldsymbol{V}_c$ can be calculated from the speed of joints via a body Jacobian Matrix $\boldsymbol{J}_b \in \mathbb{R}^{6\times 6}$:

$$\boldsymbol{V}_c = \boldsymbol{J}_b(\boldsymbol{\Theta})\dot{\boldsymbol{\Theta}} \tag{4}$$

where $\boldsymbol{V}_c = [\boldsymbol{\omega}_c; \boldsymbol{v}_c] \in \mathbb{R}^{6\times 1}$, $\boldsymbol{\omega}_c \in \mathbb{R}^{3\times 1}, \boldsymbol{v}_c \in \mathbb{R}^{3\times 1}$ are the angular velocity and the linear velocity of $\boldsymbol{P}_c$ expressed in frame $\{c\}$, respectively, and $\boldsymbol{P}_c$ is the point that coincides with the origin of frame $\{c\}$. The body Jacobian matrix $\boldsymbol{J}_b$ can be calculated by:

$$\boldsymbol{J}_b(\boldsymbol{\Theta}) = [Ad_{T_{sc}}]^{-1} \boldsymbol{J}_s(\boldsymbol{\Theta}) \tag{5}$$

where $[Ad_{T_{sc}}] \in \mathbb{R}^{6\times 6}$ is the adjoint representation of $\boldsymbol{T}_{sc}$, $\boldsymbol{J}_s$ is the space Jacobian matrix that can be obtained by the screw axis list $\boldsymbol{S}$ and the joint coordinate $\boldsymbol{\Theta}$. It is worth noting that the RCM constraint does not need to be considered in the modeling, as the laparoscope remains stationary in the x and y directions at the RCM.

2) Joint-to-tendon Kinematics: The distribution of tendons is illustrated in Fig. 2(c). The red tendons (0 and 9-th) are symmetrically placed to control the rotational joint of the camera, while the blue ones (5, 11, and 17-th) control the universal joint U_2, and the green ones (4, 10, and 16-th) control the universal joint U_1. In Fig. 2(b), the frames $\{m\}$ and $\{n\}$ are established at the two ends of each universal joint, respectively. The disk with frame $\{n\}$ is closer to the camera. The z-axes of both frames are perpendicular to the corresponding disk, and the x-axes point to the 0-th hole. The j-th hole is numbered counterclockwise around the disk, starting from the 0-th hole, with $\varphi = j\pi/9$. The coordinates of the j-th hole in the corresponding disk coordinate frame are as follows:

$$\boldsymbol{o}_j = [r_{dis}\cos(\varphi); r_{dis}\sin(\varphi); 0] \tag{6}$$

where r_{dis} is the distribution radius of the tendon holes within a disk.

The length of one tendon passing through the j-th hole between two disks is:

$$L_{m,n,j} = \|\boldsymbol{T}_{mn}[\boldsymbol{o}_j; 1] - [\boldsymbol{o}_j; 1]\| \tag{7}$$

where $\boldsymbol{T}_{mn} \in SE(3)$ describes the configuration relation of frame $\{n\}$ with respect to frame $\{m\}$:

$$\boldsymbol{T}_{mn}(\boldsymbol{\Theta}_{mn}) = e^{[\boldsymbol{S}_{m,n,1}]q_{m,n,1}} e^{[\boldsymbol{S}_{m,n,2}]q_{m,n,2}} \boldsymbol{M}_{mn} \tag{8}$$

where $\boldsymbol{M}_{mn}$ is the initial configuration of frame $\{n\}$ expressed in the frame $\{m\}$, $\boldsymbol{\Theta}_{mn} = [q_{m,n,1}; q_{m,n,2}]$ is the vectorized joint coordinate and $\boldsymbol{S}_{m,n,1}, \boldsymbol{S}_{m,n,2}$ are corresponding screw axes. As shown in Fig. 2(c), when the rotational joint moves, the change in the red tendon satisfies:

$$\Delta L_{r,0} = -\Delta L_{r,9} = q_6\sqrt{r_h^2 + p_h^2} \tag{9}$$

where r_h and $2\pi p_h$ are the radius and pitch of the helical groove of the rotating part.

Considering the effect of the translational joint upon all tendons, the change in tendon lengths during laparoscopic motion can be therefore obtained by:

$$\Delta L_j = \begin{cases} \Delta L_t + \Delta L_{m_{U_1},n_{U_1},j} + \Delta L_{m_{U_2},n_{U_2},j} + \Delta L_{r,j}, j = 0,9 \\ \Delta L_t + \Delta L_{m_{U_1},n_{U_1},j} + \Delta L_{m_{U_2},n_{U_2},j}, j = 5,11,17 \\ \Delta L_t + \Delta L_{m_{U_1},n_{U_1},j}, j = 4,10,16 \end{cases} \tag{10}$$

where $\Delta L_t = q_1$ is the change in tendon length caused by translational joint, and $\Delta L_{m_{U_1},n_{U_1},j}, \Delta L_{m_{U_2},n_{U_2},j}$ are the change in the j-th tendon length caused by universal joint U_1, U_2, respectively.

3) Mapping from Tendon Space to Driving Space: As shown in the motor driving module in Fig. 1, all tendon ends are fixed on connectors linked to sliders. The motor rotation is converted into linear slider translation through the lead screw mechanism, achieving a change in tendon length. The conversion formula between the change in tendon length and the motor rotation angle is:

$$\theta = \Delta L / l \times 360^\circ \tag{11}$$

where θ is the rotation angle of the motor and l is the lead of the lead screw structure.

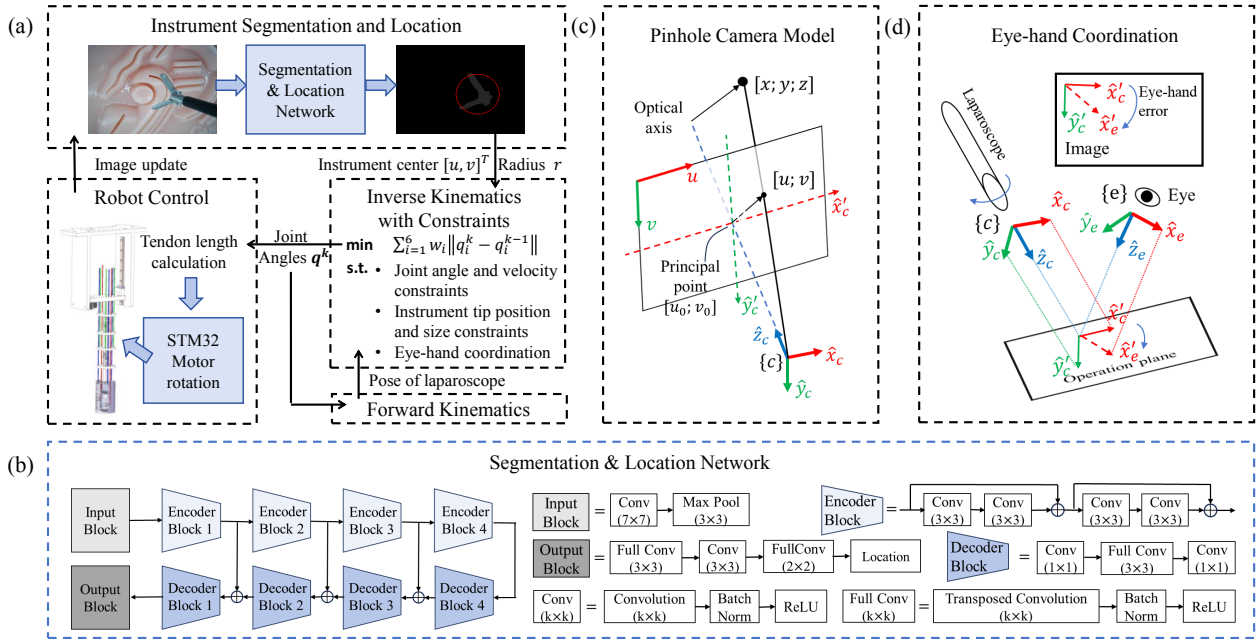

Fig. 3. (a) The general framework of the automatic FOV control system. (b) Structure of the instrument tip segmentation and localization network. (c) Schematics of pinhole camera model. (d) Schematics of eye-hand error.

III. Automatic Control of Tendon-driven Continuum Laparoscope

The general framework of the laparoscopic automatic control system is shown in Fig. 3(a). This section first introduces the segmentation and localization method of surgical instrument tips and then presents an image-based visual servo model for the proposed tendon-driven continuum laparoscope.

A. Instrument Segmentation and Location using Deep Learning Method

Recently, deep neural networks with encoder-decoder architecture have been widely used to perform image semantic segmentation tasks. In this work, the LinkNet architecture is adopted as the backbone and modified, as shown in Fig. 3(b), to achieve real-time segmentation of surgical instrument tips because LinkNet demonstrates an excellent balance between accuracy and real-time performance [14]. Additionally, the dataset SITSSD [8], which consists of 1874 surgical instrument images, is divided into a training set and a validation set with an 80:20 ratio, specifically for network training. After 50 generations of training, the mean Intersection over Union (mIoU) achieved on the validation set is 95.04%.

The segmentation results are subsequently used to extract central position $[u;v]$ and size information r of the instrument tip in the image. It is important to note that there are two types of pixels in the segmentation annotation: "tip" and "background". The central position $[u;v]$ is the center of all "tip" class pixels. To determine the size information, the distance from all "tip" class pixels to the central position is arranged in ascending order, and the distance corresponding to the 95th percentile is designated as the radius r. This approach aims to reduce the impact of misclassified pixels. The processing speed of the entire instrument segmentation and location process on a single NVIDIA GeForce GTX 1660 Super is 28.25 frames per second.

B. Image-based Visual Servo Control of Tendon-driven Continuum Laparoscope

1) Image-based Visual Servo Control Model: Since a micro-camera is used, the pinhole camera model is adopted as the imaging model in this paper, as shown in Fig. 3(c). The relationship between the central position of an instrument tip in 3D space $\boldsymbol{P} = [x;y;z]$ and its image projection $\boldsymbol{p} = [u;v]$ is:

$$s[\boldsymbol{p};1] = \boldsymbol{A}[\boldsymbol{R}_{cs}|\boldsymbol{t}_{cs}][\boldsymbol{P};1] \tag{12}$$

with

$$\boldsymbol{A} = \begin{bmatrix} f_u & 0 & u_0 \\ 0 & f_v & v_0 \\ 0 & 0 & 1 \end{bmatrix} \tag{13}$$

where s is an arbitrary scale factor, $\boldsymbol{A}$ is the matrix of intrinsic parameters, $[\boldsymbol{R}_{cs}|\boldsymbol{t}_{cs}]$ are the extrinsic parameters, which relate the world coordinate frame to the camera coordinate frame, $\boldsymbol{R}_{cs} \in SO(3)$ is the rotation and $\boldsymbol{t}_{cs} \in \mathbb{R}^{3\times1}$ is the translation. f_u, f_v are the focal lengths expressed in pixel units and $[u_0;v_0]$ is the principal point.

With the camera model developed above, the image-based visual servo model can be obtained by:

$$s\boldsymbol{v}_{uv} = \boldsymbol{J}\boldsymbol{v}_c \tag{14}$$

where $\boldsymbol{v}_{uv} = [\dot{u};\dot{v}]$ is velocity of image projection of an instrument tip center $\boldsymbol{p}$ and $\boldsymbol{J} = \begin{bmatrix} -f_u & 0 & u-u_0 \\ 0 & -f_v & v-v_0 \end{bmatrix}$. Eq.(14) presents the relationship between the linear velocity of $\boldsymbol{P}_c$ and that of $\boldsymbol{p}$.

Let $\boldsymbol{J}_{b,v} = \boldsymbol{J}_b(4:6,:)$ and substitute Eq.(4) to Eq.(14):

$$s\boldsymbol{v}_{uv} = \boldsymbol{J}\boldsymbol{J}_{b,v}(\boldsymbol{\Theta})\dot{\boldsymbol{\Theta}} \tag{15}$$

In the proposed laparoscopic control method, the product of $s\boldsymbol{v}_{uv}$ is a discrete variable, determined by pixel error ($\boldsymbol{PE}$):

$$s\boldsymbol{v}_{uv}^k = K_p\widehat{\boldsymbol{PE}}^k + K_i\sum \alpha\widehat{\boldsymbol{PE}}^k + K_d(\widehat{\boldsymbol{PE}}^k - \widehat{\boldsymbol{PE}}^{k-1}) \tag{16}$$

where $\boldsymbol{v}_{uv}^k$ is the velocity of $\boldsymbol{p}^k = [u^k; v^k]$ at time step k. $\widehat{\boldsymbol{PE}}^k = \boldsymbol{PE}^k/\|\boldsymbol{PE}^k\|$ and $\boldsymbol{PE}^k = \boldsymbol{p}^k - \boldsymbol{p}_{des}$ is the pixel error at time step k, $\boldsymbol{p}_{des} = [u_{des}; v_{des}]$ is the desired central position of an instrument tip in the image. K_p, K_i, K_d are the gains of the PID controller and $\alpha \in (0,1]$ is the decay factor of the integral process.

During the FOV automatic adjustment, both the central position and the size of an instrument tip in the image, described by r, need to be considered. The size of the instrument tip in the image is negatively correlated with the distance from the camera's end to the instrument tip. Therefore, r can be adjusted by changing the linear velocity in the z-direction of the camera's end, denoted as $z_{\boldsymbol{v}_c}$. Let $\boldsymbol{J}_{b,v_z} = \boldsymbol{J}_{b,v}(3,:)$, then:

$$z_{\boldsymbol{v}_c} = \boldsymbol{J}_{b,v_z}(\boldsymbol{\Theta})\dot{\boldsymbol{\Theta}} \tag{17}$$

Similar to $s\boldsymbol{v}_{uv}$, $z_{\boldsymbol{v}_c}$ is a discrete variable, determined by the radius error (RE):

$$z_{\boldsymbol{v}_c}^k = K_{p,r}RE^k + K_{d,r}(RE^k - RE^{k-1}) \tag{18}$$

where $RE^k = r^k - r_{des}$ is the radius error at time step k, r^k is the radius represent for size information and r_{des} is the desired radius. $K_{p,r}, K_{d,r}$ are the gains of the PD controller.

2) Eye-hand Coordination: The difference between the laparoscope's line-of-sight and the surgeon's natural line-of-sight when looking directly into the operation plane impairs eye-hand coordination, as illustrated in Fig. 3(d). This misorientation makes it challenging to carry out complex procedures successfully. Eye-hand error is defined as the angle difference between the projection direction of the x-axis of the camera frame $\{c\}$ and the x-axis of the eye frame $\{e\}$ on the operation plane. To eliminate this eye-hand error, it is necessary to rotate the camera at the end of the laparoscope so that the projection directions of the x-axes of both frames onto the operation planes are collinear [15]. That is, the x-axis of frame $\{c\}$ should be on the plane determined by the z-axis of frame $\{c\}$ and the x-axis of frame $\{e\}$, and point towards the positive direction of the x-axis of frame $\{e\}$.

The normal vector of the plane determined by the z-axis of frame $\{c\}$ and the x-axis of frame $\{e\}$ is:

$$\boldsymbol{n} = \widehat{\boldsymbol{x}}_e \times \boldsymbol{R}_{ec}(\boldsymbol{\Theta})\widehat{\boldsymbol{z}}_c \tag{19}$$

where $\boldsymbol{R}_{ec} \in SO(3)$ is the rotation matrix that represents the orientation of frame $\{c\}$ relative to frame $\{e\}$, $\widehat{\boldsymbol{x}}_e = [1;0;0]$ and $\widehat{\boldsymbol{z}}_c = [0;0;1]$ are the x-axis of frame $\{e\}$ and the z-axis of frame $\{c\}$, respectively. The x-axis of frame $\{c\}$ expressed in frame $\{e\}$ is:

$$\widehat{\boldsymbol{x}}_{ec} = \boldsymbol{R}_{ec}(\boldsymbol{\Theta})\widehat{\boldsymbol{x}}_c \tag{20}$$

where $\widehat{\boldsymbol{x}}_c = [1;0;0]$ is the x-axis of frame $\{c\}$. The angle difference between $\widehat{\boldsymbol{x}}_{ec}$ and $\boldsymbol{n}$ is:

$$\beta(\boldsymbol{\Theta}) = \begin{cases} \text{atan2}(-\|\boldsymbol{n}\times\widehat{\boldsymbol{x}}_{ec}\|, \boldsymbol{n}^T\widehat{\boldsymbol{x}}_{ec}), & \boldsymbol{z}_{\boldsymbol{n}\times\widehat{\boldsymbol{x}}_{ec}} > 0 \\ \text{atan2}(\|\boldsymbol{n}\times\widehat{\boldsymbol{x}}_{ec}\|, \boldsymbol{n}^T\widehat{\boldsymbol{x}}_{ec}), & \boldsymbol{z}_{\boldsymbol{n}\times\widehat{\boldsymbol{x}}_{ec}} \le 0 \end{cases} \tag{21}$$

where $\beta \in (-\pi, \pi]$ and $\boldsymbol{z}_{\boldsymbol{n}\times\widehat{\boldsymbol{x}}_{ec}}$ is the third dimension of $\boldsymbol{n}\times\widehat{\boldsymbol{x}}_{ec}$. Furthermore, the eye-hand error θ_{err} can be calculated by:

$$\theta_{err}(\boldsymbol{\Theta}) = \begin{cases} \beta(\boldsymbol{\Theta}) - 3\pi/2, & \beta \in (\pi/2, \pi] \\ \beta(\boldsymbol{\Theta}) + \pi/2, & \text{otherwise} \end{cases} \tag{22}$$

The eye-hand coordination satisfies well when θ_{err} is close to 0.

3) Inverse Kinematics with Constraints: To meet the constraints derived above while minimizing the joint motion, the objective function about the joint coordinate $\boldsymbol{\Theta}^k$ is set as:

$$f(\boldsymbol{\Theta}^k) = \sum_{i=1}^{6} w_i \|q_i^k - q_i^{k-1}\| \tag{23}$$

where q_i^k is the joint angle of the i-th joint at time step k, and w_i is the weight factor for the i-th joint.

The optimization object of solving the inverse kinematics with constraints can be summarized as follows:

$$\begin{aligned} \min \quad & f(\boldsymbol{\Theta}^k) \\ \text{s.t.} \quad & \boldsymbol{\Theta}^{min} \le \boldsymbol{\Theta}^k \le \boldsymbol{\Theta}^{max} \\ & \left\| \boldsymbol{\Theta}^k - \boldsymbol{\Theta}^{k-1} \right\| \le \Delta\boldsymbol{\Theta}^{max} \\ & s\boldsymbol{v}_{uv}^k = \boldsymbol{J}\boldsymbol{J}_{b,v}(\boldsymbol{\Theta}^k)(\boldsymbol{\Theta}^k - \boldsymbol{\Theta}^{k-1})/dt \\ & z_{v_c}^k = \boldsymbol{J}_{b,v_z}(\boldsymbol{\Theta}^k)(\boldsymbol{\Theta}^k - \boldsymbol{\Theta}^{k-1})/dt \\ & \theta_{err}(\boldsymbol{\Theta}^k) = 0 \end{aligned} \tag{24}$$

Various algorithms exist to solve the above quadratic programming, such as MATLAB function *fmincon*. The resolved joint angles are then used to calculate the tendon lengths and the corresponding motor rotations for controlling the laparoscope.

IV. Simulation

Two simulation platforms are developed under MATLAB and Webots environments to intuitively test and optimize the proposed automatic FOV adjustment algorithm.

TABLE I
PARAMETER SETTINGS FOR MATLAB SIMULATIONS.

Parameter	Value	Parameter	Value
Z_0 (mm)	100	f_u (pixel)	500
L (mm)	49.88	f_v (pixel)	500
L_{cam} (mm)	61.94	u_0, v_0 (pixel)	320,240
r_{dis} (mm)	11.5	r_{des} (pixel)	100
r_h (mm)	4	u_{des}, v_{des} (pixel)	320,240
$2\pi p_h$ (mm)	3	w	1,3,3,1,1,1
K_p	75	$[q_1^{min}, q_1^{max}]$ (mm)	[20,100]
K_i	30	$[q_i^{min}, q_i^{max}]$, i=2,3,4,5	[−35°,35°]
K_d	20	$[q_6^{min}, q_6^{max}]$	[−180°,180°]
α	0.8	Δq_i^{max} (rad), i=2,3,4,5	0.02
$K_{p,r}$	3	$\boldsymbol{R}_{se}$	$diag([1,-1,-1])$
$K_{d,r}$	0.1	Image Size (pixel)	640 × 480

A. Simulation on MATLAB

A simulated continuum laparoscopic robot is developed in MATLAB. Detailed parameters, including the robot's specifications, controller gains, and camera parameters, are shown in Table I. The initial joint angles are set as $\boldsymbol{\Theta}^1 = [20;0;0;0;0;0.3]$. Three simulation cases, comprising one static case and two dynamic cases, are carried out in MATLAB. The trajectories of each case are as follows:

- Case "Point" (static):

$$\boldsymbol{P}(t) = [50;30;0] \qquad \boldsymbol{P}_{cir}(t) = [60;30;0]$$

- Case "Line3D" (dynamic):

$$\boldsymbol{P}(t) = [-20;0;0] + ([100;0;50] - [-20;0;0]) \times t/T$$

$$\boldsymbol{P}_{cir}(t) = \boldsymbol{P}(t) + [0;10;0]$$

- Case "Circle3D" (dynamic):

$$\boldsymbol{P}(t) = [50\cos(\omega t);50\sin(\omega t);10+20\sin(\omega t)]$$

$$\boldsymbol{P}_{cir}(t) = [60\cos(\omega t);60\sin(\omega t);10+20\sin(\omega t)]$$

where $\boldsymbol{P}, \boldsymbol{P}_{cir}$ are the central position and the edge point of the instrument tip, and the distance between them is the radius of the instrument tip. The units of coordinates are mm. t is current time, $\omega = 2\pi/T$, and $T = 30s$ is the total simulation time.

The simulation results of the three cases are displayed in Fig. 4. Figs. 4(a) - (c) show the central position of the instrument tip in the image, where the red dots represent the tracked results and the green dots represent the untracked results. It is evident that without the proposed tracking method, the instrument eventually moves to the edge or even out of the laparoscope's FOV. Figs. 4(d) - (f) display the curves of pixel error, instrument tip radius in the image, and eye-hand error against time, respectively.

The initial values of pixel errors for the cases "Point", "Line3D", and "Circle3D" are 291.55, 100.00, and 277.78 pixels, with initial radii of 50.00, 50.00, and 55.56 pixels, respectively. The initial eye-hand error for all cases is -17.19° (-0.3 rad). It can be observed from Fig. 4 that in the case of "Point", the central position and the size of the instrument tip in the image reach the target state within about 2s, indicating that the center of the instrument tip is located at the center of the image, and the radius is 100 pixels.

In the two dynamic simulation cases, the central position and the size of the instrument tip in the image tend to converge after 1s, ensuring eye-hand consistency during the FOV adjustment process. The simulation statistics of the two dynamic cases after 1s are shown in Table II. For the "Line3D" case and the "Circle3D" case, the maximum errors are 5.49 and 20.45 pixels, respectively, indicating that the instruments are located near the center of the image throughout the entire tracking process. Additionally, for the "Line3D" case, the maximum and minimum radii after stable tracking are 100.52 pixels and 99.79 pixels, while for the "Circle3D" case, the maximum and minimum radii after stable tracking are 100.81 pixels and 98.87 pixels. The radius

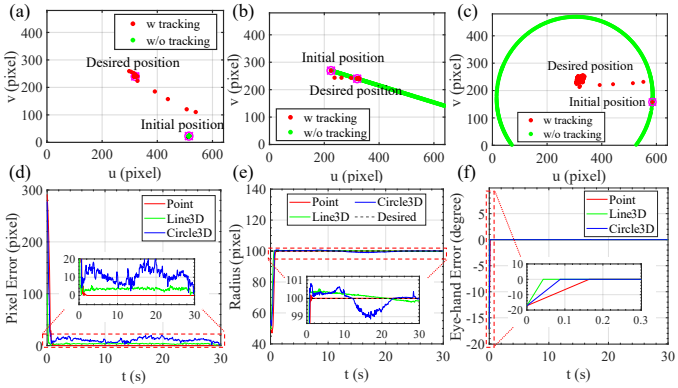

Fig. 4. MATLAB Simulation results. (a) - (c) Trajectory of the instrument tip center in the image with and without tracking: (a) "Point", (b) "Line3D", (c) "Circle3D". (d) Tracking error vs. time. (e) Instrument tip radius vs. time. (f) Eye-hand error vs. time.

TABLE II

SIMULATION RESULTS OF DYNAMIC CASES ON MATLAB (AFTER 1S)

		Line3D	Circle3D
Pixel Error (pixel)	Max	5.4872	20.4497
	Mean	3.4555	9.9723
Radius (pixel)	Max	100.5216	100.8110
	Min	99.7936	98.8684
Eye-hand Error (degree)	Max (abs)	4.1330e-04	4.1100e-02
	Mean (abs)	1.0568e-06	7.9635e-05

varies slightly around the expected value, with the maximum error value not exceeding 1.50 pixels. Once convergence is reached, the eye-hand error approaches 0, with a maximum error of only 0.04°. The simulation results of the above three evaluation criteria are competitive to a recent study [8], where a rod-shaped laparoscope is used. Specifically, the average tracking error is 15.46 pixels, the average absolute radius error is 1.54 pixels, and the average absolute eye-hand error is 0.17° when tracking a single instrument moving along a 3D circular trajectory in MATLAB simulation.

B. Simulation on Webots

To intuitively and visually study and optimize the visual servo control algorithm, the proposed continuum laparoscopic robot is developed in the Webots environment, as shown in Fig. 5. A simplified model of the continuum laparoscopic robot, consisting of a base, three links, two universal joints, and a camera, is created as shown in Fig. 5(b). To simulate a surgical instrument that can move freely in space, a small car with a ball on top is designed (Fig. 5(c)). The car can move freely on a horizontal plane, while the ball can move vertically relative to the car. This setup simulates the random movements of a surgeon manipulating surgical instruments during a real surgical procedure. Lastly, the enclosure fence represents the abdominal cavity, and the range of motion for the ball is confined within it.

The *recognition* function provided by the Webots camera is used to identify the target, i.e., the ball, as shown in Fig. 5(d). The position and size of the target can be obtained in real-time. The *keyboard* function offered by Webots allows users to control the random movement of the target point

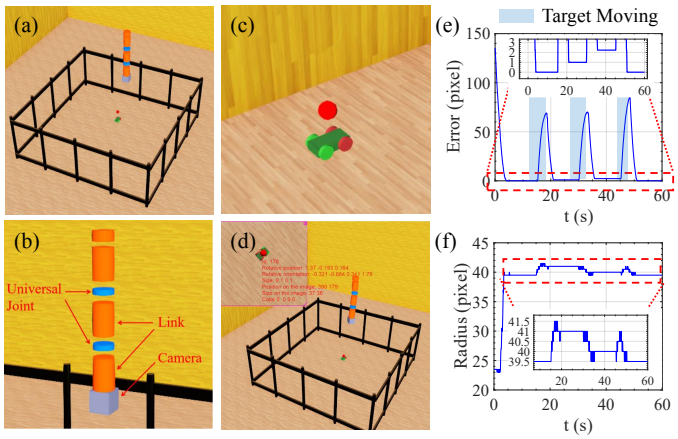

Fig. 5. Simulation on Webots. (a) Simulation environment setup. (b) Simplified continuum laparoscope robot. (c) The tracking target. (d) Recognition results of the tracking target. (e) Tracking error vs. time. The movement periods of the target are highlighted in blue. (f) Radius of target vs. time. The desired radius is 40 pixels.

in 3D space. During the simulation, the coordinates and size of the target in the camera frame are extracted. Then, the aforementioned algorithm calculates the joint angles of the laparoscope robot. Eventually, the robot adjusts its configuration to ensure the target is tracked within the center of the camera's FOV and remains an appropriate size. To minimize screen shaking, the algorithm stops adjusting the configuration of the robot once the tracking error is below 10 pixels. Simulation results of tracking error and radius of the target in the image are shown in Fig. 5(e) and (f), respectively. The target is randomly moved at 15s, 30s, and 45s by an operator via keyboard, and the movement periods are highlighted in blue. It is evident that the tracking error decreases to less than 3 pixels within 5s. Meanwhile, the radius of the target in the image converges to the desired value of 40 pixels within 5s. And the max radius error after convergence is 1.5 pixels. The simulation process can be seen in the supplementary video.

V. Prototype Validation

A. Experiment Setup

As shown in Fig. 6(a), the experiment setup includes a tendon-driven continuum laparoscope and an MIS abdominal cavity model. The tendon-driven continuum laparoscope robot has a total length of 223.76 mm and a maximum radius of 14 mm. The rotation angle range of each universal joint is [-35°, 35°]. The laparoscope robot is equipped with a miniature camera installed at the end with a resolution of 640×480. It communicates with the host computer through a USB port at a rate of up to 30 Hz. The laparoscope camera is calibrated using Zhang's calibration method [16]. The camera's internal parameters are $f_u = 573.61, f_v = 574.42, u_0 = 289.00, v_0 = 243.48$, ignoring camera distortion. The chessboard pattern is used for eye-hand consistency calibration. Figs. 6(b) and (c) show the images captured by the camera and from the outside of the MIS abdominal cavity model during the experimental process, respectively.

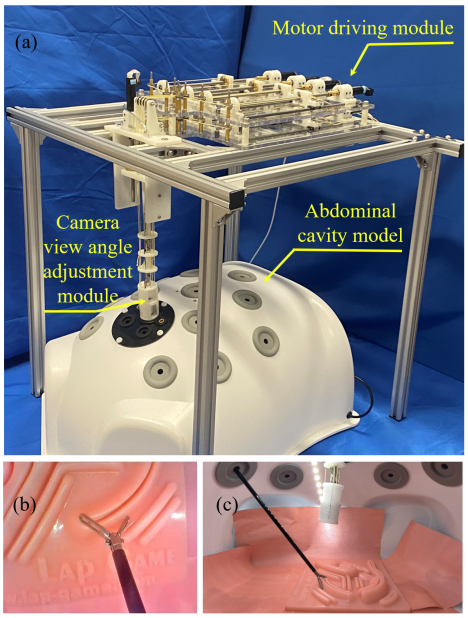

Fig. 6. The experimental platform. (a) The device overview. (b) The image captured by the end camera during the experiment. (c) Internal view of the abdominal cavity model.

B. Accuracy Verification for Instrument Tracking

In the experiment of FOV automatic adjustment, one MIS instrument is randomly moved to different positions by an operator at 15s, 30s, and 45s, while it remains roughly stationary for the rest of the time to simulate the movement of instruments during the surgical process. Fig. 7(a) shows representative positions of a surgical instrument on the screen at 4 time steps: 0s, 10s, 19s, and 60s. Initially, the surgical instrument is located in the upper left corner of the camera's FOV. Then, the laparoscope automatically tracks it to center it in the FOV, achieving automatic adjustment of the surgical FOV. At 19s, the surgical instrument is moved to the upper right corner of the FOV. The proposed system takes roughly 4s to adjust the FOV. The tracking error of a single test is shown in Fig. 7(b), where the highlighted blue areas represent the periods of sudden movement of a surgical instrument. The experimental results show that the proposed robotic system, along with the FOV adjustment algorithm, can effectively track the moving instrument, with an average convergence time of about 4s and an average tracking error of about 15 pixels during convergence.

The changes in tendon lengths during the instrument tracking process are shown in Fig. 8. The results indicate that the proposed tendon-driven continuum laparoscope robot can achieve the desired configuration within an acceptable error range. The experimental process can be found in the supplementary video.

VI. Conclusion

This paper proposes a new type of tendon-driven continuum laparoscopic robot, which avoids the influence of RCM during the FOV adjustment process. Based on this laparoscope, this paper comprehensively considers the position and

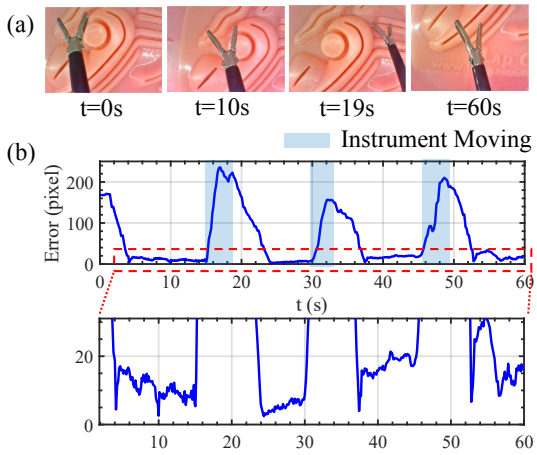

Fig. 7. (a) Screenshots of the tracking process at 4 time steps. (b) Tracking error vs. time for a single instrument tracking. The movement periods of the surgical instrument are highlighted in blue.

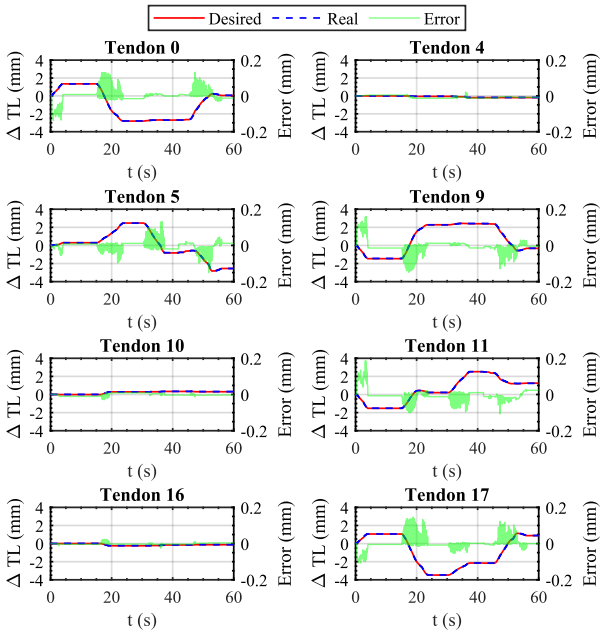

Fig. 8. Changes in tendon lengths during one instrument tracking process. TL stands for tendon length.

size of a surgical instrument in the image as well as eye-hand consistency to design an automatic visual field adjustment algorithm. A deep learning-based method is used for real-time instrument localization, achieving marker-free tracking of the instrument. The feasibility of the proposed automatic laparoscopic FOV adjustment system is verified through simulations and prototype experiments. The experimental results show that the proposed system can adjust laparoscopic FOV effectively, with a convergence time of about 4s and an average tracking error of about 15 pixels during stabilization. However, there is still certain room for improvement. For instance, the prototype of the laparoscopic robot is manufactured using laboratory rapid prototyping methods, and further optimization of its size is needed. Future work also includes incorporating advanced dynamic model-based control methods for tendon-driven robots developed by the group [17] to improve control accuracy.

REFERENCES

[1] M. Kim, Y. Zhang, and S. Jin, "Soft tissue surgical robot for minimally invasive surgery: a review," *Biomedical Engineering Letters*, vol. 13, no. 4, pp. 561–569, Nov. 2023.

[2] W. Wang, Y. Luo, J. Wang, X. Wang, and H. Song, "Robotic-Assisted Laparoscopic Adjustment: A Meta-Analysis and Review," *IEEE Transactions on Instrumentation and Measurement*, vol. 72, pp. 1–19, 2023.

[3] T. Osa, C. Staub, and A. Knoll, "Framework of automatic robot surgery system using Visual servoing," in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct. 2010, pp. 1837–1842.

[4] T. Mahmood, S. W. Cho, and K. R. Park, "DSRD-net: Dual-stream residual dense network for semantic segmentation of instruments in robot-assisted surgery," *Expert Systems with Applications*, vol. 202, p. 117420, Sep. 2022.

[5] L. Sestini, B. Rosa, E. De Momi, G. Ferrigno, and N. Padoy, "FUN-SIS: A fully UNsupervised approach for surgical instrument segmentation," *Medical Image Analysis*, vol. 85, p. 102751, Apr. 2023.

[6] C. Gruijthuijsen, L. C. Garcia-Peraza-Herrera, G. Borghesan, D. Reynaerts, J. Deprest, S. Ourselin, T. Vercauteren, and E. Vander Poorten, "Robotic Endoscope Control Via Autonomous Instrument Tracking," *Frontiers in Robotics and AI*, vol. 9, 2022.

[7] K. Fozilov, J. Colan, A. Davila, K. Misawa, J. Qiu, Y. Hayashi, K. Mori, and Y. Hasegawa, "Endoscope Automation Framework with Hierarchical Control and Interactive Perception for Multi-Tool Tracking in Minimally Invasive Surgery," *Sensors*, vol. 23, no. 24, p. 9865, Dec. 2023.

[8] C. Zhang, W. Zhu, J. Peng, Y. Han, and W. Liu, "Visual servo control of endoscope-holding robot based on multi-objective optimization: System modeling and instrument tracking," *Measurement*, vol. 211, p. 112658, Apr. 2023.

[9] X. Ma, X. Wang, R. Cao, and K. W. S. Au, "Design, Teleoperation Control and Experimental Validation of a Dexterous Robotic Flexible Endoscope for Laparoscopic Surgery," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct. 2022, pp. 5937–5944.

[10] J. W. Lee, S. H. Choi, S. Kim, and S. W. Kwon, "Laparoscopic liver resection for segment VII lesion using a combination of rubber band retraction method and flexible laparoscope," *Surgical Endoscopy*, vol. 34, no. 2, pp. 954–960, Feb. 2020.

[11] Z. Wang, S. Bao, B. Zi, Z. Jia, and X. Yu, "Development of a Novel 4-DOF Flexible Endoscopic Robot Using Cable-Driven Multisegment Continuum Mechanisms," *Journal of Mechanisms and Robotics*, vol. 16, no. 031011, Mar. 2023.

[12] X. Zhang, W. Li, W. Y. Ng, Y. Huang, Y. Xian, P. W. Yan Chiu, and Z. Li, "An Autonomous Robotic Flexible Endoscope System with a DNA-inspired Continuum Mechanism," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, May 2021, pp. 12 055–12 060.

[13] Y. Huang, W. Li, X. Zhang, J. Li, Y. Li, Y. Sun, P. W. Y. Chiu, and Z. Li, "4-DOF Visual Servoing of a Robotic Flexible Endoscope With a Predefined-Time Convergent and Noise-Immune Adaptive Neural Network," *IEEE/ASME Transactions on Mechatronics*, pp. 1–12, 2023.

[14] A. Chaurasia and E. Culurciello, "LinkNet: Exploiting encoder representations for efficient semantic segmentation," in *2017 IEEE Visual Communications and Image Processing (VCIP)*, Dec. 2017, pp. 1–4.

[15] S. Yang, Y. Wang, H. Zhao, H. Cheng, and H. Ding, "Autonomous Laparoscope Control for Minimally Invasive Surgery With Intuition and RCM Constraints," *IEEE Robotics and Automation Letters*, vol. 7, no. 3, pp. 7934–7941, Jul. 2022.

[16] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 11, pp. 1330–1334, Nov. 2000.

[17] Q. Chen, M. Li, H. Wu, W. Liu, and J. Peng, "Design, self-calibration and compliance control of modular cable-driven snake-like manipulators," *Mechanism and Machine Theory*, vol. 193, p. 105562, Mar. 2024.