

Surgical Suture Thread Detection and 3-D Reconstruction Using a Model-Free Approach in a Calibrated Stereo Visual System

Bo Lu^{id}, Henry K. Chu^{id}, Member, IEEE, Kaicheng Huang^{id}, and Jiewen Lai^{id}

Abstract—In robot-assisted surgery (RAS), three-dimensional (3-D) position estimation of a surgical suture thread is essential in automating the wound suturing procedure. Nevertheless, accurate suture thread's segmentation and its 3-D coordinates reconstruction remain challenging because of its flexible property and lacking indistinctive feature points. In this paper, a novel model-free method is proposed to enable the 3-D computation of a suture thread through a calibrated stereo camera system. In this method, only the suture thread's tip and grasping points need to be indicated on one image with mouse clicks, and their accurate image positions can be detected using computed contour information. Afterward, key points that represent the suture thread are generated on one image, while their matched points on the second camera frame are evaluated through an iterative algorithm for 3-D model reconstruction. Experiments using different backgrounds and environmental conditions were conducted to examine the accuracy and robustness of the 2-D segmentation. The suture threads were then formed in various 3-D shapes, and their respective 3-D models were reconstructed for comparison. The experimental results confirm the feasibility of our method in reducing human intervention in the detection of matched key points on an image pair, thereby guaranteeing an accurate 3-D coordinates computation of a suture thread.

Index Terms—Computer vision for medical robotics, perception for grasping and manipulation, stereo camera.

I. INTRODUCTION

ROBOT-ASSISTED surgery has flourished over the recent decades [1]. With the aid of haptic and other types of sensory feedback [2], [3], surgical robots can perform dexterous tasks [4] with high repeatability. Similar to traditional surgery,

Manuscript received March 31, 2019; revised August 22, 2019; accepted September 16, 2019. Date of publication September 20, 2019; date of current version April 15, 2020. This work was supported in part by the Research Grant Council of the Hong Kong Special Administrative Region, China, under Grant 25204016 and in part by the Hong Kong Polytechnic University, under Grant G-YBXR. Recommended by Technical Editor X. Chen. This work was supported in part by the Research Grant Council of the Hong Kong Special Administrative Region, China, under Grant 25204016 and in part by the Hong Kong Polytechnic University, under Grant G-YBXR. (*Corresponding author: Henry Chu.*)

The authors are with the Department of Mechanical Engineering, The Hong Kong Polytechnic University, Hong Kong (e-mail: bo.lu@connect.polyu.hk; henry.chu@polyu.edu.hk; huangkc0907@hotmail.com; jw.lai@connect.polyu.hk).

Color versions of one or more of the figures in this article are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TMECH.2019.2942715

knot tying is one of the most common and repetitive tasks to be performed in RAS. In particular for technically demanding applications such microvascular anastomosis [5] and reconstructive pelvic procedure [6], knot tying needs to be performed with high precision in a limited space. Hence, researchers are studying the potential to automate these procedures by using vision-based robots [7].

To complete autonomous surgical tasks such as knot tying, it is necessary to utilize an advanced robotic system to detect and grasp the suture thread in a surgery scenario. Kang *et al.* [8] presented a novel robotic system with multiple programmable devices and interchangeable end tools to facilitate an autonomous suturing. Jackson *et al.* [9] proposed an algorithm for needle path planning based on the best practices of surgeons and showed that their proposed method can minimize the interaction forces between the tissue and the needle when suturing. Sen. *et al.* [10] proposed a multithrow stitching method based on sequential convex programming that uses a Three-dimensional (3-D) printed suture needle angle positioner. These works focused on the development of novel methods and paradigms related to the stitching process, which is the initial step of autonomous knot tying.

In the field of robotic knot tying, Chow *et al.* [11], [12] proposed a roll-arc looping method to construct knots with visual guidance. Lu *et al.* [13] proposed a space-saving method to loop a suture thread in a compact environment using visual information [14]. To assist surgeons in tying a surgical knot, autonomous manipulation of suture thread grasping is required so that the autonomous looping operation can be executed. In general, a surgical suture consists of two components: a sharp metal needle on the proximal end of the suture, and a thread that can be used to close up the wound. After successful stitching, a scissor is used to cut the long thread, leaving a segment of the thread that is sufficient to complete a surgical knot. After the cutting operation, the suture thread will drop arbitrarily on the skin. As shown in Fig. 1, the cutting tip is denoted as P_T , and P_G is the grasping point. The main objective is to facilitate the grasper in picking up the suture thread. The distance between P_G and P_T is reserved as a surgical “ear” [15] to ensure that the suture thread will not slip after its grasping. To automate this operation, precise 3-D computation on a segment of a long and flexible suture thread is required.

Traditional object detection methods [16]–[21] require specific patterns, computer-aided design (CAD) models, fea-

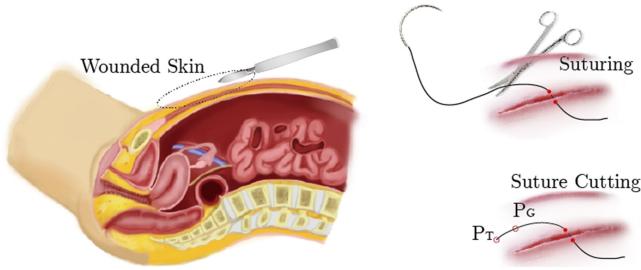


Fig. 1. Cutting of a suture thread after the pierce operation, and its defined tip P_T and the grasping point P_G .

tures, detectors, or local descriptors to perform the recognition procedure. However, suture threads have no representative patterns or explicit features that can be extracted in different orientations and scales, thereby preventing the generation of invariant local descriptors. Considering these factors, these typical methods are not suitable for surgical suture thread detection.

Given these challenges, learning-based approaches have been introduced for feature detection [22]. In [23], discriminative convolutional filters using a Gradient Boosting framework were employed for segmentation of a curvilinear structure. However, the accuracy of this method drops when the linear structures become very irregular. A learning-based approach was also proposed in [24] for detecting symmetric curvilinear objects in natural images based on the extracted features. Multiple instance learning was also employed to train the detector, but poor accuracy in detecting suture threads with limited features is still unsolved. With these uncertainties, it was clarified in [25] that allowing robots to take exploratory actions in real-world environments may pose safety and stability problems, and some execution failures can still be observed when using deep learning models or when learning by demonstration. Thus, a shared autonomy with initial manual guidance [26] is desirable for precise manipulation.

Deformable curvilinear objects, such as threads and cables, can be also evaluated by using an energy model based on a texture-sensitive distance map in [27]. Alternatively, suture threads were also modeled as nonuniform Rational B-spline curves and optimized via the minimization of the image match energy [28], [29]. When using a model to represent the suture thread, its quality and feasibility will heavily influence the tracking results.

In practice, robotic systems play a supporting role by helping surgeons perform various surgical procedures. Through captured images, surgeons can interact with the system by issuing simple commands or clicking the mouse. To compute 3-D coordinates of an object, the current state-of-art is using a stereo camera system [30], and such paradigm is also adopted in robot-assisted surgery [31], [32]. With such a device, the 3-D shape can be figured out based on object's key point correspondences in both cameras.

The first step in 3-D computation is to identify the matched points on the image pair from a stereo camera. Nevertheless, extracting features from a flexible and featureless suture thread is a challenging task as common operators such as scale-invariant

feature transform (SIFT) are unable to generate key points on a featureless object [33]. Markers can be added to alleviate the problem [34], but this option requires modifications to the standard suture thread. With insufficient number of key points to be extracted from the images, 3-D position evaluation as well as shape reconstruction of the suture thread become inaccurate.

To resolve these problems and realize vision-based automated suture thread's grasping, a novel model-free approach named as the initial-position-driven iterative computation (IPdIC) algorithm was proposed. Through a precalibrated stereo camera, surgeons only need to select the tip P_T and the grasping point P_G of the suture thread in the left camera frame. With this input, a robust shared autonomy scheme can be initialized. The program first applies a novel edge filter to outline the contour of the suture thread. Then, our algorithm refines the position errors of the manual clicks, and key points between P_T and P_G will be generated in the left frame. Then, the actual fundamental matrix at the tip is computed by using the chessboard calibration data and will be iteratively updated along the segmented suture's contour such that the matched key points in the right camera can be identified and optimized for 3-D position reconstruction. This algorithm can be applied for all suture threads regardless of their image scale, orientation, length, and visual depth.

This paper is organized as follows. In Section II, the proposed method for identifying the tip of the suture thread will be introduced. In Section III, the key points generation and refinement algorithms, as well as the iterative approach for computing their stereo correspondences will be presented. In Section IV, the performance for tip detection, key point generation, stereo matching, and 3-D reconstruction will be examined through experiments. The feasibility of the proposed approach will also be validated by testing the 3-D construction and computation. In Section V, findings of this paper will be discussed and summarized.

II. SURGICAL SUTURE THREAD'S TIP DETECTION

A. Three-Layer Contour Segmentation of a Suture Thread

To initialize the 3-D computation of a suture thread, the first step is to find its tip location P_T in image coordinates. To achieve this, the image should be converted to a high-quality contour pattern. The edge or contour is usually computed by calculating the derivatives in the horizontal and vertical directions. The Sobel operator is a popular method for estimating such derivatives [35]. Developed by Canny in 1986 [36], the function optimization detector was improved in [37] by adding a scale multiplication to the function.

Definition 1: Using image processing operators, the camera frame can be converted to a binary image \mathcal{B} . We have the following:

- 1) \mathcal{P}_i is the image location of pixel $i \rightarrow \mathcal{P}_i = [U_i, V_i]$;
- 2) $\tilde{\mathcal{P}}_i$ denotes the intensity of pixel i in binary image \mathcal{B} ;
- 3) centered at point \mathcal{P}_i , G_i indicates a 3×3 local patch;
- 4) the contour set is denoted by $\mathbf{P}_{1,E}$, which contains the coordinates and intensity information. The subscript $\{1, E\}$ means the first to the E th pixel in the set.

5) for the convenience of expression, coordinate notation \mathcal{P}_i is used to denote pixel i in the following content.

To segment the desired suture thread's contour $\mathbf{P}_{1,E}$, the edge detection algorithm should satisfy two criteria.

- 1) A continuous contour should be generated, and it should be one-pixel-wide and no zigzag patterns.
- 2) $\forall \mathcal{P}_i \in \mathbf{P}_{1,E}$, the neighboring number of \mathcal{P}_i in its local patch G_i should be 2.

Since classical methods cannot fully satisfy these criteria, a three-layer contour segmentation algorithm was proposed. This approach starts with a *local scanning operation*, which follows a row-by-row fashion, to evaluate the intensity value $\tilde{\mathcal{P}}_i$ of each \mathcal{P}_i . The intensity values can be updated as

$$\tilde{\mathcal{P}}_i \leftarrow \tilde{\mathcal{P}}_i \cdot \left[1/e^{(\tilde{\mathcal{P}}_{i,h-\min} - \tilde{\mathcal{P}}_{i,v-\min})} \right]. \quad (1)$$

where $\tilde{\mathcal{P}}_{i,h-\min}$ and $\tilde{\mathcal{P}}_{i,v-\min}$, respectively, denote the minimal pixel intensity among the horizontal and vertical neighbors of \mathcal{P}_i , which are

$$\begin{cases} \tilde{\mathcal{P}}_{i,h-\min} = \min\{\tilde{\mathcal{P}}(U_i, V_i - 1), \tilde{\mathcal{P}}(U_i, V_i + 1)\} \\ \tilde{\mathcal{P}}_{i,v-\min} = \min\{\tilde{\mathcal{P}}(U_i - 1, V_i), \tilde{\mathcal{P}}(U_i + 1, V_i)\}. \end{cases} \quad (2)$$

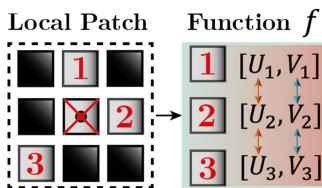
For $\forall \mathcal{P}_i \in \mathcal{B}$, if the updated intensity $\tilde{\mathcal{P}}_i$ is equal to *one*, it will be treated as a temporary contour element and recorded in contour set $\mathbf{P}_{1,E}$. As shown in Fig. 2 (the first *layer*), a region of interest (RoI) was drawn around the tip for the evaluation. With the implementation of (1), pixels within the suture thread were removed, resulting a rough contour.

In the second *layer*, the remaining members in $\mathbf{P}_{1,E}$ will be examined whether they are redundant or not using a local patch evaluator. For any 3×3 patch G_i , it can be converted into two patterns using the following operators:

$$\mathbb{O}_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad \mathbb{O}_2 = {}^T \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix} \quad (3)$$

where T is the matrix transposition operator. \mathbb{O}_1 and \mathbb{O}_2 mean the pattern at the same and rotated 90° clockwise, respectively. As illustrated in Fig. 2—the second layer, each pattern was evaluated using a function f , which is

Definition 2: Using a *local scanning operation*, all nonzero pixels except at the center are selected and sequentially aligned in a set. For any pixel within this set, if differences between the image coordinates U and V of the current and its adjacent elements are neither larger than 1, its output will be 0, otherwise the returned value will be 1.



The pseudo codes of Function f are listed in Algorithm 1. Implementing the double-channel evaluator, the intensity of

Algorithm 1: Local Patch Evaluator $f(\mathcal{P}_i, G_i)$.

```

1 Current point  $\mapsto \mathcal{P}_i = [U_i, V_i]$ , the intensity value  $\tilde{\mathcal{P}}_i$ ,
   local patch  $\mapsto G_i, j = 1, Value = 0$ ;
2 while  $j \leq 9$  do
3   if  $\forall \mathcal{P}_j \in G_i \wedge \mathcal{P}_j \neq \mathcal{P}_i$  then
4     if  $\tilde{\mathcal{P}}_j = 1 \wedge j \geq 2$  then
5       if  $(U_j - U_{j-1} \geq 1) \parallel (U_j - U_{j-1} \geq 1)$  then
6          $Value = Value + 1$ ;
7   if  $Value = 1$  then
8     Break;
9 Return  $Value$  of the algorithm.

```

each member in $\mathbf{P}_{1,E}$ can be modified as

$$\forall \mathcal{P}_i \in \mathbf{P}_{1,E} \quad \tilde{\mathcal{P}}_i \leftarrow \tilde{\mathcal{P}}_i \cdot \prod_{j=1}^2 f(\mathcal{P}_i, G_i \cdot \mathbb{O}_j) \quad (4)$$

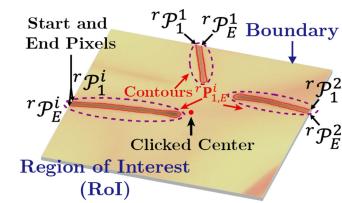
where $G_i \cdot \mathbb{O}_j$ denotes the transformed local patch centered at \mathcal{P}_i using operators in (3). If the pixel intensity satisfies $\tilde{\mathcal{P}}_i = 0$, the point \mathcal{P}_i will be removed from $\mathbf{P}_{1,E}$.

To obtain a satisfactory contour pattern in any operating scenario, the third *layer* was designed to iteratively evaluate the difference of contour pixels between the current filtered image and the previous one before feeding to the second layer. A superscript i is added to $\mathbf{P}_{1,E}$ to denote the contour in the i th iteration. Feed the current image to the second layer, if the difference $\Delta^i \mathbf{P}$ between ${}^i \mathbf{P}_{1,E}$ and ${}^{i+1} \mathbf{P}_{1,E}$ is nonzero, the iteration should be continued, otherwise, the procedure should be terminated, and the output pattern should be the current obtained image, which fully satisfies the two criteria.

B. Identification of Suture Thread's Tip Point in RoI

With the modified camera frame, a RoI can be generated based on the clicked thread's tip P_T . In different operating scenarios, disturbances or noises may exist that multiple $\mathbb{C}_{\text{RoI}} = \{{}^r \mathbf{P}_{1,E}^1, {}^r \mathbf{P}_{1,E}^2, \dots, {}^r \mathbf{P}_{1,E}^i, \dots\}$ can be obtained. The left superscript indicates the domain is in the RoI, and the right one denotes the number of the found candidates. For the desired suture thread within RoI, it holds these properties:

Property 1: There exists at least one candidate $\exists {}^r \mathbf{P}_{1,E}^i \in \mathbb{C}_{\text{RoI}}$, it is a continuous contour and holds two ends (start and end) ${}^r \mathcal{P}_1^i$ and ${}^r \mathcal{P}_E^i$ on the boundary \mathbb{B}_{RoI} of the RoI.



Property 2: The pixel distance ${}^r d_1^i$ between ${}^r \mathcal{P}_1^i$ and ${}^r \mathcal{P}_E^i$ should be smaller than a threshold. If it exceeds it, the candidate may be a large shadow or long thick disturbances.

Property 3: The desired ${}^r \mathbf{P}_{1,E}^i$ should contain the largest pixel number, and the distance ${}^r d_2^i$ between its recognized tip ${}^r \mathcal{P}_t^i$ and the clicked point should be close.

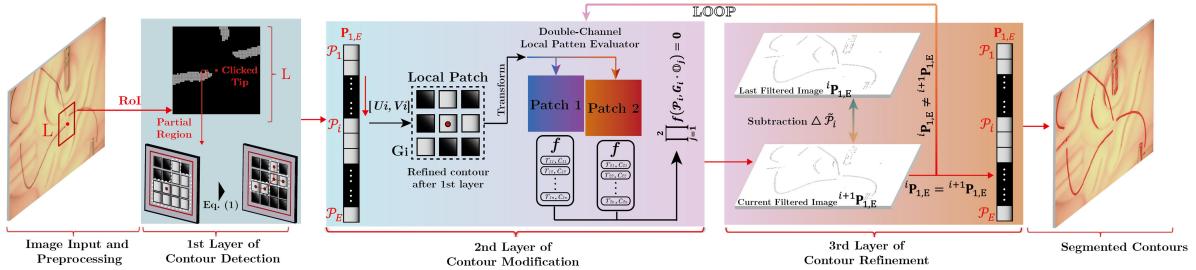


Fig. 2. Workflow of the three-layer contour segmentation and thinning approach of the interested objects within the camera frame. 1) The first layer computes the rough contour of the interested object within the frame. 2) The second layer modifies the contour based on the proposed double-channel evaluation criteria. 3) The last layer compares the current filtered image with the one obtained in the last loop, and obtains perfect mono-layer and closed object contours.

The tip holds the property of the sharpest changes among the curve within the RoI. For each candidate $r\mathcal{P}_{1,E}^i$, one pixel was picked out in every η_1 numbers (we set $\eta_1 = 8$). The vectors $r\vec{\mathcal{V}}_{k-1,k}^i$ and $r\vec{\mathcal{V}}_{k,k+1}^i$ can be figured out based on three successive $r\mathcal{P}_{k-1}^i$, $r\mathcal{P}_k^i$, and $r\mathcal{P}_{k+1}^i$. Consequently, the angle variance along the contour can be expressed as

$$r\Delta\theta_k^i = \arccos \frac{r\vec{\mathcal{V}}_{k-1,k}^i \cdot r\vec{\mathcal{V}}_{k,k+1}^i}{|r\vec{\mathcal{V}}_{k-1,k}^i| \cdot |r\vec{\mathcal{V}}_{k,k+1}^i|}. \quad (5)$$

The tip $r\mathcal{P}_t^i$ of each candidate can be achieved by figuring out the image point where the largest $r\Delta\theta_k^i$ happens within the RoI, and the corresponding distance $r d_2^i$ that between $r\mathcal{P}_t^i$ and the clicked point can thereby be accessed.

To distinguish the desired suture thread, a penalty function ν^i was formulated according to the three properties, and disturbances could be given larger punishments

$$\nu^i = e^{\frac{r d_1^i}{p_1} - 1} + p_2 * r d_2^i + \frac{p_3}{r n^i} \quad (6)$$

where the suture thread's width $r d_1^i$ can be calculated as $r d_1^i = ||r\mathcal{P}_1^i - r\mathcal{P}_E^i||$, and p_1 is its weighting parameter. p_2 denotes the weighting parameter of the error between the clicked point and the detected tip of the candidate, and p_3 is the weight of the reciprocal of the contour element number $r n^i$ in $r\mathcal{P}_{1,E}^i$. If the candidate holds a smaller $r d_1^i$ value, and its recognized tip is closer to the clicked point, as well as having a larger number of contour elements, it obtains a smaller penalty. Thus, the desired set of the suture thread $r\mathcal{P}_{1,E}^S$ within the RoI can be picked out as

$$r\mathcal{P}_{1,E}^S \in \mathbb{C}_{\text{RoI}} \rightarrow \nu^S = \min\{\nu^1, \nu^2, \dots, \nu^i, \dots\}. \quad (7)$$

Consequently, the suture thread's tip $r\mathcal{P}_t^S$ in the RoI can be achieved, and its coordinates $\mathcal{P}_t^S = [U_t^S, V_t^S]$ among the entire image can be located. Searching the connecting point from the tip point, its complete contour $\mathbf{P}_{1,E}^S$ can be sequentially figured out until the search finally returns to the tip, forming a closed outline as shown in Fig. 3.

The abovementioned method applied to segment the desired suture thread from the camera frame is named as click-based contour identification and closure (CCIC) Algorithm, and its corresponding pseudo codes are summarized in Algorithm 2.

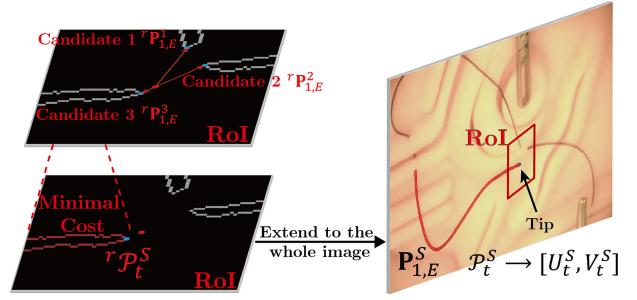


Fig. 3. Suture thread's tip selection based on the minimal cost evaluation with in the RoI. The tip \mathcal{P}_t^S and its overall contour $\mathbf{P}_{1,E}^S$ can be figured out in the camera frame.

Algorithm 2: Click-based Contour Identification and Closure (CCIC) Algorithm

Data: The binary image \mathcal{B} after image preprocessing; Clicked tip location $[U_c, V_c]$ of the suture thread.

- 1: $\forall \mathcal{P}_i \in \mathcal{B} \xrightarrow[\text{1st Layer}]{\text{Eq.(1) (Eq.(2)}} \text{ evaluate local patch } G_i \rightarrow$ updated intensity image $\tilde{\mathcal{P}}_i$ and the contour set $\mathbf{P}_{1,E}$;
- 2: $\forall \mathcal{P}_i \in \mathbf{P}_{1,E} \rightarrow$ apply double-channel ((3)) patch evaluator $\xrightarrow[\text{2nd Layer}]{\text{Eq.(4)}}$ obtain a new contour set $i+1\mathbf{P}_{1,E}$;
- 3: Compare the current $i+1\mathbf{P}_{1,E}$ with the last one $i\mathbf{P}_{1,E}$, if $\Delta \mathbf{P}$ is not null, go to Step 2; otherwise, go to Step 4;
- 4: Create RoI \rightarrow contours $r\mathcal{P}_{1,E}^i$ of multiple candidates $\xrightarrow{\text{Eq.(5)}}$ recognize tips of candidates $r\mathcal{P}_t^i$;
- 5: Compute the penalty ν^i of each candidate $\xrightarrow{\text{Eq.(6)(7)}}$ find the minimal one \rightarrow treat it as the desired suture thread;

Result: Desired suture thread $\mathbf{P}_{1,E}^S$ and its tip $\mathcal{P}_t^S = [U_t^S, V_t^S]$.

III. COMPUTATION OF SUTURE THREAD'S KEY POINTS IN A STEREO CAMERA SYSTEM

The 3-D computation of the suture thread should be further proceeded using a calibrated stereo camera. In the following contents, the left and right cameras are labeled with the right superscripts ℓ and r .

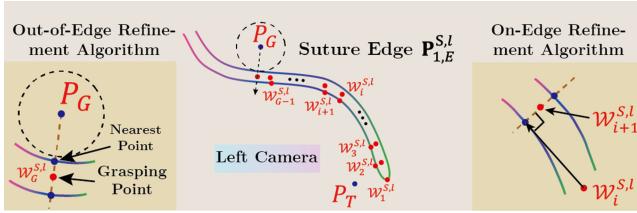


Fig. 4. Suture thread's key points planning in the left camera.

A. Key Points Generations in 2-D Frame

With the segmented contour $\mathbf{P}_{1,E}^{S,\ell}$ in the left camera, the clicked grasping point P_G can be refined regarding errors of manual clicks. The objective of the refinement is to shift the key points to suture's centreline, and two refinement algorithms regarding two initial conditions were proposed.

Algorithm 3: Out-of-Edge Refinement: Suppose one point \mathbf{P}_i is not located on the contour, the procedures are:

- 1). Create a region of interest based on point \mathbf{P}_i ;
- 2). Figure out one point \mathbf{P}_j from the suture thread's contour which is nearest to \mathbf{P}_i , and draw a line across \mathbf{P}_i and \mathbf{P}_j ;
- 3). Figure out another intersection point \mathbf{P}_k between this line and the contour, the refinement is $\mathbf{P}_i \leftarrow \frac{1}{2} \cdot (\mathbf{P}_i + \mathbf{P}_k)$.

As noticed in Fig. 4, the click grasping point G_p can be updated using this scheme, which is denoted as $\mathcal{W}_G^{S,\ell}$.

A suture thread can be modeled as a multisegment object, and it is connected by equally distributed key points. To further generate its key points, we picked out contour pixels from $\mathbf{P}_{1,E}^{S,\ell}$ in every η_2 elements (we set $\eta_2 = 25$ according to the hardware setup). Let the detected suture thread's tip be the first key point $\mathcal{W}_1^{S,\ell}$ in the left camera, the picked out pixels can be refined using.

Algorithm 4: On-Edge Refinement: Suppose one point \mathbf{P}_i is located on the thread's contour, the procedures are:

- 1). Create a region of interest based on point \mathbf{P}_i ;
- 2). Generate a line using the last key point and the current point \mathbf{P}_i , and compute the line's slope. Generate a second line which is across \mathbf{P}_i and orthogonal to the first line.
- 3). Figure out the another intersection \mathbf{P}_j between the second line and the suture thread's contour, and \mathbf{P}_i can be refined as $\mathbf{P}_i \leftarrow \frac{1}{2} \cdot (\mathbf{P}_i + \mathbf{P}_j)$.

As shown in Fig. 4, successive pixels can be selected from the contour set, and then shifted to its body's center.

Stop criteria: When modifying the manual-clicked grasping point P_G , two intersection points can be figured out, and their marshallling sequence \mathcal{I}_1 and \mathcal{I}_2 within the set $\mathbf{P}_{1,E}^{S,\ell}$ can be thereby obtained. With a current selected contour element whose marshallling sequence is \mathcal{I}_c , the planning should be terminated if it satisfies $|\mathcal{I}_c - \mathcal{I}_1| < \eta_2$ or $|\mathcal{I}_c - \mathcal{I}_2| < \eta_2$. Therefore, a set

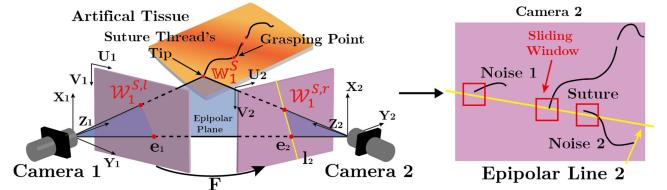


Fig. 5. Epipolar line based searching and matching of the suture thread's tip within a calibrated stereo camera system.

of 2-D key points $\mathbf{W}_{1,G}^{S,\ell} = \{\mathcal{W}_1^{S,\ell}, \mathcal{W}_2^{S,\ell}, \dots, \mathcal{W}_G^{S,\ell}\}$ can be achieved in the left camera. This key points planning method based on the segmented contour can also be applied to an overlapped or self-intersected suture thread, ensuring a reliable robustness for its 3-D reconstruction and the automated grasping task.

B. Evaluations of Suture Thread's Tip Correspondences

According to the epipolar geometry [38] of a stereo camera as shown in Fig. 5, the line formed by two centers of the cameras intersects at two epipoles e_1 and e_2 at their image frames. The projection of the 3-D suture thread's tip \mathbb{W}_1^S in the right camera projection $\mathcal{W}_1^{S,r}$ should lie on the epipolar line l_2 , which can be computed as

$$l_2 = \mathbf{F} \cdot \mathcal{W}_1^{S,\ell} \quad (8)$$

where \mathbf{F} is the fundamental matrix of the stereo camera.

To find the desired suture thread in the right camera, a sliding window was designed to move along the epipolar line and figure out the potential candidates. Within the sliding window, different contours were recognized, and thereby different tips can be achieved. Based on this information, the overall contours $\mathbf{P}_{1,E}^{i,r}$ of different candidates can be figured out. Following the maximum similarity principle, the Pearson correlation coefficient [39] between $\mathbf{P}_{1,E}^{i,r}$ and the segmented suture thread's contour $\mathbf{P}_{1,E}^{S,\ell}$ can be computed as

$$\text{Cor}(i) = \frac{\sum_{k=1}^E (\mathcal{P}_k^{i,r} - \bar{\mathcal{P}}_{1,E}^{i,r})(\mathcal{P}_k^{S,\ell} - \bar{\mathcal{P}}_{1,E}^{S,\ell})}{\sqrt{\sum_{k=1}^E (\mathcal{P}_k^{i,r} - \bar{\mathcal{P}}_{1,E}^{i,r})^2 \cdot \sum_{k=1}^E (\mathcal{P}_k^{S,\ell} - \bar{\mathcal{P}}_{1,E}^{S,\ell})^2}} \quad (9)$$

where the subscript k specifies each single point in its contour set, $\bar{\mathcal{P}}_{1,E}^{S,\ell}$ and $\bar{\mathcal{P}}_{1,E}^{i,r}$ denote the mean coordinate values of contour elements of the suture thread in the left camera and the i th candidate found based on the sliding-window searching strategy in the right camera. The candidate with a maximum $\text{Cor}(i)$ can be selected as the correct suture thread $\mathbf{P}_{1,E}^{S,r}$, and the tip recognized in the sliding window can be treated as the desired tip $\mathcal{P}_t^{S,r}(\mathcal{W}_1^{S,r})$ in the right camera.

C. Maximum Virtual Triangle Based Iterative Computation for Key Points Detection

For a fixed stereo camera system, calibration can be performed offline using a chessboard with known dimensions. By

placing the chessboard at different orientations and depths, camera parameters can be obtained [40], [41] and 3-D coordinates can be calculated using the corresponding point pair in the two cameras. In this paper, with markerless key points generated along the suture in the left camera, an iterative approach was used to find their corresponding points in the right camera using the following approach.

Proof: For a 3-D point \mathbf{X}_i of an object, its perspective projection onto a camera image \mathbf{x}_i can be computed as

$$\mathbf{s}_i \cdot [\mathbf{x}_i, 1]^T = \mathbf{P} \cdot [\mathbf{X}_i, 1]^T = \mathbf{C} \cdot [\mathbf{R}, \mathbf{T}] \cdot [\mathbf{X}_i, 1]^T \quad (10)$$

where s_i is the nonzero scale parameters, $\mathbf{P} \in \mathbb{R}^{3 \times 4}$ denotes the projection matrix. $\mathbf{C} \in \mathbb{R}^{3 \times 3}$ is a camera calibration matrix, and $\mathbf{R} \in \mathbb{R}^{3 \times 3}$ and $\mathbf{T} \in \mathbb{R}^{3 \times 3}$ represent its rotation and translation matrices with respect to the world coordinate. When the distance of the object from the camera is much larger than its depth variation, the perspective model can be simplified to an affine model, and the projection matrix is

$$\mathbf{P} = \begin{bmatrix} P_{11} & P_{12} & P_{13} & P_{14} \\ P_{21} & P_{22} & P_{23} & P_{24} \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{A} & \bar{\mathbf{T}} \\ \mathbf{0}^T & 1 \end{bmatrix} \quad (11)$$

where $\mathbf{A} \in \mathbb{R}^{2 \times 3}$ is the submatrix of \mathbf{P} , and $\bar{\mathbf{T}}$ is a 2-D translation vector. Within a stereo vision system, the 3-D point and its projections onto the left and right cameras after removing the scale factor s_i can be rewritten as

$$\begin{cases} \bar{\mathbf{x}}_i^\ell = \mathbf{A}^\ell \cdot \bar{\mathbf{X}}_i + \bar{\mathbf{T}}^\ell \\ \bar{\mathbf{x}}_i^r = \mathbf{A}^r \cdot \bar{\mathbf{X}}_i + \bar{\mathbf{T}}^r \end{cases} \rightarrow \begin{bmatrix} \bar{\mathbf{x}}_i^\ell \\ 1 \end{bmatrix} = \underbrace{\begin{bmatrix} \mathbf{K}^{\ell \rightarrow r} & \mathbf{T}^{\ell \rightarrow r} \\ 0 & 1 \end{bmatrix}}_{\text{Affine}} \cdot \begin{bmatrix} \bar{\mathbf{x}}_i^r \\ 1 \end{bmatrix} \quad (12)$$

where $\mathbf{K}^{\ell \rightarrow r} = \mathbf{A}^r \cdot [\mathbf{A}^\ell]^T \in \mathbb{R}^{2 \times 2}$, $\mathbf{T}^{\ell \rightarrow r} = \mathbf{A}^r \cdot [\{\mathbf{A}^\ell\}^{-1} \cdot \mathbf{T}^\ell] \in \mathbb{R}^{2 \times 2}$. It is noticed the projection correspondences in the left and right cameras follow an affine transformation, which has six unknown parameters. ■

Corollary: For a 3-D plane π that is formed by three points $\{\mathbf{W}_a^\pi, \mathbf{W}_b^\pi, \mathbf{W}_c^\pi\}$, their projections onto the left and right cameras $\{\mathcal{W}_a^\ell, \mathcal{W}_b^\ell, \mathcal{W}_c^\ell\}$ and $\{\mathcal{W}_a^r, \mathcal{W}_b^r, \mathcal{W}_c^r\}$ can fix an affine transformation \mathcal{H}^π [42], [43]. For point \mathbf{W}_d^π that is close to any of the three points, its projections \mathcal{W}_d^ℓ and \mathcal{W}_d^r follow the same affine relationship.

Before the on-site manipulation, sufficient stereo correspondences of key points among a chess board were acquired and denoted as $\mathbf{V}^C = \{\mathbf{V}_1^C, \mathbf{V}_2^C, \dots, \mathbf{V}_i^C, \dots, \mathbf{V}_j^C, \dots\}$.

Iteration: Treating them as 3-D virtual points, one virtual plane can be created using the suture thread's tip \mathbf{W}_1^S and two virtual points \mathbf{V}_i^C and \mathbf{V}_j^C , as shown in Fig. 6. Applying the affine relationship \mathcal{H}_1 obtained using their projections in two cameras, together with the planned 2-D key point $\mathcal{W}_2^{S,\ell}$, the stereo correspondence $\mathcal{W}_2^{S,r}$ can be accessed. Using these latest key point pairs, a new plane is created to update the affine transformation. Following such iterations, the suture thread's key points $\{\mathcal{W}_1^{S,r}, \mathcal{W}_2^{S,r}, \dots, \mathcal{W}_G^{S,r}\}$ can be obtained.

To obtain a robust affine relationship between two frames, projections of three points should be noncollinear, otherwise the

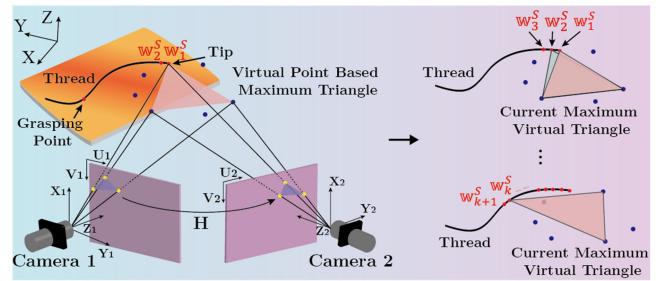


Fig. 6. Iterative computation of stereo pairs of the suture thread's key points based on maximum triangle.

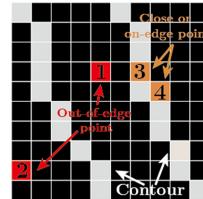
computation of the key point may have severe errors, disordering following calculations.

Update 1: To achieve three noncollinear points, a maximum virtual triangle principle was forwarded. For a current key point \mathcal{W}_k^S , every two random virtual points $\mathcal{V}_i^C, \mathcal{V}_j^C$ were picked out from \mathbf{V}^C , and 2-D triangles' areas in the left and right cameras, which were formed by projections $\{\mathcal{W}_k^{S,\ell}, \mathcal{V}_i^{C,\ell}, \mathcal{V}_j^{C,\ell}\}$ and $\{\mathcal{W}_k^{S,r}, \mathcal{V}_i^{C,r}, \mathcal{V}_j^{C,r}\}$ were both taken into account. For two different virtual points, the evaluation coefficient notated as $\alpha_{\{i,j\},k}$ can be expressed as

$$\alpha_{\{i,j\},k} = \prod_{\Phi=1}^2 (\mathcal{V}_i^{C,\Phi} - \mathcal{W}_k^{S,\Phi}) \times (\mathcal{V}_j^{C,\Phi} - \mathcal{W}_k^{S,\Phi}) \quad (13)$$

where $\Phi = \{1, 2\}$ denote the left and right camera, respectively. The subscript i, j are the numbers of two evaluated virtual points, and k means the k th key point pairs. In each iteration, two virtual points were updated based on the principle of maximizing the evaluation coefficient. With an updated relationship, the next key point pairs can be figured out. To further refine key points' coordinates and precisely modify them to the suture thread's centreline, a two-conditions-based update scheme was adopted in each iteration.

Update 2: As shown in figure below, if the distance between the predicted $\mathcal{W}_k^{S,r}$ and the contour is larger than 1 pixel (Conditions 1 and 2), apply Algorithm 3: Out-of-Edge Refinement to rectify its position. Otherwise (Conditions 3 and 4), rectify their positions using Algorithm 4.



Consequently, the key point correspondences $\mathbf{W}_{1,G}^{S,r} = \{\mathcal{W}_1^{S,r}, \mathcal{W}_2^{S,r}, \dots, \mathcal{W}_G^{S,r}\}$ in the right camera can be achieved, and their 2-D accuracies were significantly improved in view of these enhancements. The suture thread's key point correspondences, as well as the calibrated parameters of the stereo cameras, were employed to compute its 3-D coordinates based

on the *triangulation* relationship [38], [44]. Connecting and smoothing them in sequence, the spatial shape of the suture can be constructed in the camera coordinate.

The abovementioned approach for suture thread's 3-D reconstruction is named as the IPdIC algorithms. Using this approach, the useful information can be completely segmented. Together with the update and refinement algorithms, our method eliminates the mismatch of key point pairs when conducting the 3-D coordinates calculation, minimizing computational errors resulting from the variations in visual depth in each iteration.

IV. EXPERIMENTAL RESULTS AND ANALYSIS

The experimental setup used for 3-D coordinates computation of a suture thread was shown in Fig. 7. A stereo camera was mounted at a tilted angle on the manipulating platform, and two eyepiece microscopes with resolutions of 1536×1024 pixels² were installed to capture the real-time visual feedback of the manipulating plane. Before the experiments, a nonabsorbable silk suture thread with a United States Pharmacopeia (USP) designation of 3 0 was pierced through an artificial tissue. To create different scenarios for analysis, suture threads with different lengths and backgrounds were employed to examine the feasibility and robustness of the overall approach.

A. Experimental Results of Suture Thread's Tip and Grasping Point Detection in Left Camera

Based on the ROI, the detection accuracy of the tip is important since the initial error will significantly affect the computation of the suture thread's stereo correspondence and its 3-D reconstruction in following steps. To testify the performance of the algorithm, six different scenarios were adopted, which were categorized as the 1X magnification condition, the 2X magnification scenario, the 1X and 2X magnifications with noisy suture threads and tips, the 1X magnification under the dark light and strong light.

Using two artificial tissues as the backgrounds, 20 experiments were conducted in each scenario. It was intentionally designed that large initial errors were given to manual clicked suture thread's tip P_T and its grasping point P_G . As shown in Fig. 8(a), typical results of each scenario were presented. Based on the CCIC algorithm, a closed contour of the suture thread can be segmented from different environments, and its tip point and the grasping point were precisely located regardless of environmental noises.

To quantitatively evaluate the detection accuracy toward the tip and grasping point, we manually labeled the ground truths in each experiment. In Fig. 8(b), the data that denotes clicked positions via ground truths, and detected coordinates via ground truths were summarized. It is noticed manual clicks could lead to larger errors and deviations from the ground truths, while the points detected through the algorithm can achieve high accuracy as compared to the ground truths. The errors of all clicked tip, grasping points, and their corresponding detected locations compared to the ground truths were listed in Fig. 8(c), which intuitively shows the significant improvements of accuracy. In

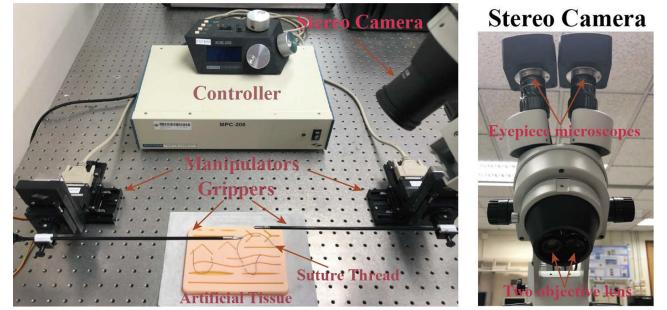


Fig. 7. Experimental system with a stereo camera for the computation of a suture thread's 3-D coordinates.

addition, the average, maximal, and minimal errors of each scenario were listed in Table I, and the largest errors of the detected tip and the grasping point were only 8.5 and 7.1 pixels among 120 experiments. It is testified that errors can be reduced to a certain level, and the proposed CCIC algorithm can be applied to initialize the suture thread's 3-D reconstruction.

With a strong robustness, the CCIC algorithm can detect the suture thread even when sudden drops happen. To approach this problem, the detected tip is treated as an updated clicked tip point for a sudden drop condition. In Fig. 9, two groups of such conditions were listed. With a grasped suture thread, the section between its tip and the grasping point can be constructed. When the suture thread drops, the overall shape enormously changed. However, the variance of tip's image location is not large. Feeding the previous tip's location as the clicked point, our approach can segment the suture thread and figure out its 3-D coordinates between P_T and P_G . It can be noticed the change of tips coordinates were 130 and 108 pixels, respectively. However, if the suture thread was severely disturbed, the change of tip's position can be very large, which demands for a new manual click to indicate its current image location.

B. Detection of Correspondences of Suture Thread's Tip and Grasping Points in the Right Camera

To testify the performance of the (IPdIC) algorithm in a suture thread's 3-D reconstruction, the matching accuracy of its key points should be examined. First, the detection accuracy of the tip in the right camera was conducted. For a fixed stereo vision system, it was designed to evaluate the robustness in four scenarios that include the normal condition, with dark light, strong light, and with noisy tips. Using a sliding window, the search of suture thread's tip in the right camera can be carried out. Based on the evaluation of the maximum similarity, the corresponding suture thread in the right camera can be figured out, and three typical segmentation examples were shown in Fig. 10(a).

Conducting 20 experiments in each scenario, errors associated with the detected tips and the manually labeled ground truths in the right camera were computed and listed in Fig. 10(b). As summarized in Table II, average errors in each scenario were all under 6 pixels images, and the maximum one was

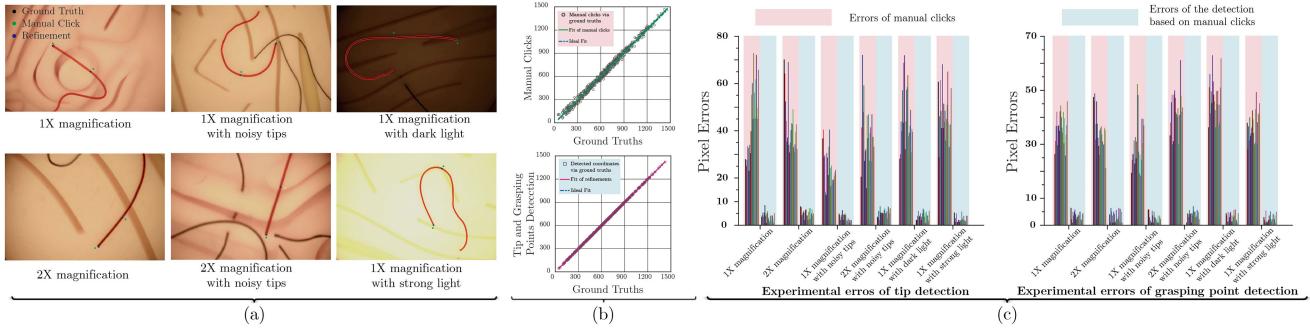


Fig. 8. (a) To testify the robustness of our approach, 6 different scenarios were adopted, and 20 experiments were conducted in each scenario. Typical experimental results of a suture thread's segmentation, its tip and grasping points detection with manual clicks were demonstrated. (b) Coordinates distributions of manual clicks via ground truths, and detected coordinates via ground truths. (c) Errors of manual clicks, and errors of corresponding tip and grasping detection. Unit: image pixel.

TABLE I

AVERAGE, MAXIMUM, AND MINIMAL ERRORS OF MANUAL CLICKS AND CLICK-BASED DETECTION OF SUTURE THREAD'S TIP AND GRASPING POINT

Scenario	1X magnification			2X magnification			Noisy tips ^a			Noisy tips ^b			Dark light ^a			Strong light ^a			
	Items	ave	max	min	Items	ave	max	min	ave	max	min	ave	max	min	ave	max	min		
Tip	Manual	45.2	72.9	23.0	43.0	70.2	30.8	25.3	40.5	12.7	38.4	72.1	15.7	45.9	72.0	28.2	47.8	68.2	28.9
	Detected	3.8	8.5	1.0	5.3	8.0	2.2	3.1	6.3	0.0	5.4	8.1	2.2	4.2	7.1	1.0	2.8	5.8	1.4
Grasp	Manual	36.3	45.9	25.8	35.4	48.8	21.2	30.1	52.3	18.4	42.3	61.1	30.0	47.3	62.9	36.2	37.1	49.3	28.3
	Detected	3.6	6.4	0.0	3.3	6.4	1.0	2.9	5.8	1.0	4.1	7.1	1.0	3.1	5.8	0.0	2.7	5.4	1.0

Two superscripts ^a and ^b indicate the 1X and 2X magnification accordingly. Unit: image pixels.

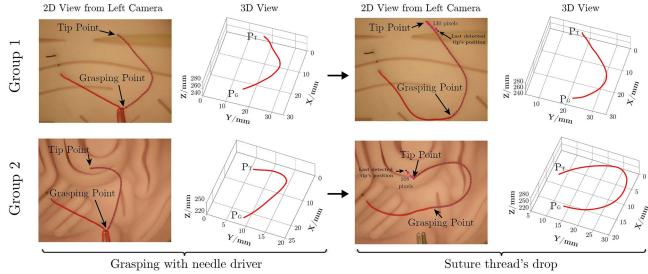


Fig. 9. Suture thread's detection with sudden drops.

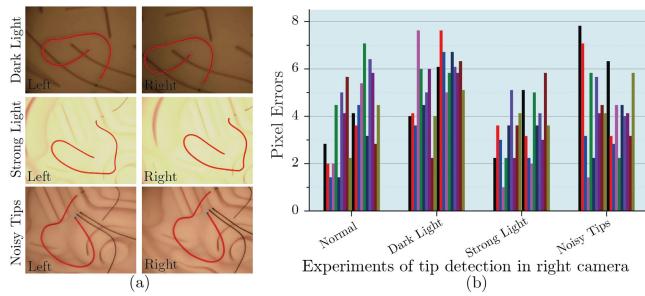


Fig. 10. Errors of the tip detection in the right camera based on manual clicks in the left camera. Unit: image pixels.

less than 8 pixels. Considering the image range, such errors are considerably small, and the accuracy is sufficiently satisfactory in the right camera's frame.

An accurate 3-D computation relies on whether the matched points were properly identified in the image pair, particularly for

TABLE II
AVERAGE, MAXIMUM, AND MINIMAL ERRORS OF DETECTED SUTURE THREAD'S TIPS IN THE RIGHT CAMERA

Items	Normal scale	Dark light	Strong light	Noisy tips
ave	3.9 pixels	5.4 pixels	3.4 pixels	4.3 pixels
max	7.1 pixels	7.6 pixels	5.8 pixels	7.8 pixels
min	1.4 pixels	2.2 pixels	1.0 pixels	1.4 pixels

the grasping point. Based on accurate pairs of suture thread's tip and other planned key points in the left camera, their correspondences can be computed by using the iterative method. To testify the accuracy of the iterative method in computing key points' stereo pairs, four different scenarios with two short and two long suture threads at different orientations were tested. A red marker was placed on the thread to represent the desired grasping point. Besides, experiments without using the refinement Algorithms 3 and 4 were also conducted as comparison.

With manually labeled grasping point in the right camera, errors between the computed coordinates and the ground truths can be obtained. For each scenario, the test was repeated $10 \times$ with various P_T and P_G positions on the image, and the typical results were shown and summarized in Fig. 11 and Table III, respectively. It can be noticed, detected key points in the right camera were also matched with the geometry of the suture thread. Compared to the results without using the refinement algorithms, errors obtained using our proposed method were all below 10 pixels and were mostly even lower than 5 pixels. With all these outcomes, it can be concluded a high accuracy of key points matching in a stereo camera can be achieved, satisfying the requirements of the following 3-D coordinates computation.

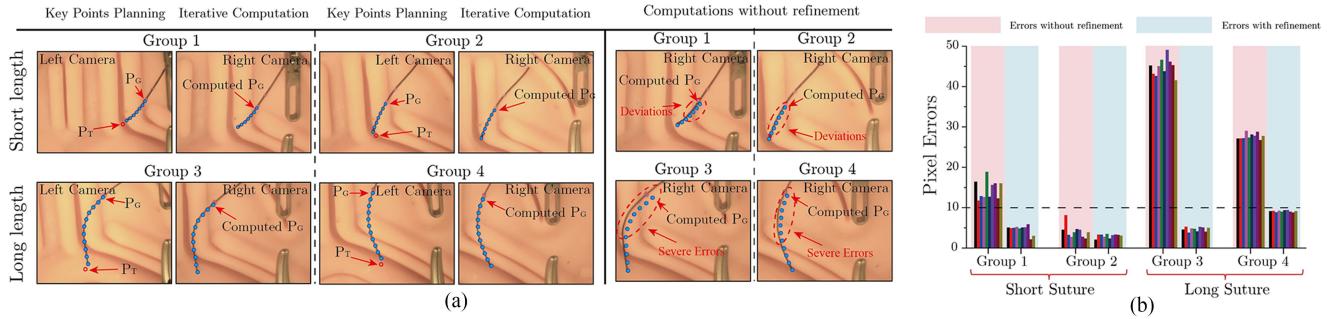


Fig. 11. Evaluation of the matching of suture's grasping point between the left and right cameras, and the importance demonstration of Algorithms 3 and 4 when computing a suture's stereo correspondences. (a) Typical examples of suture's key points generations along the suture in the left camera, and their correspondences calculations in the right camera. (b) Errors of grasping point computation with and without using Algorithms 3 and 4. Unit: image pixels.

TABLE III
AVERAGE, MAXIMUM, AND MINIMUM ERRORS OF DETECTED GRASPING POINTS IN THE RIGHT CAMERA

Items	Average		Maximum		Minimum	
	without	with	without	with	without	with
Group 1	14.5	4.6	18.9	5.9	11.7	2.1
Group 2	4.1	3.0	8.1	3.5	2.3	2.1
Group 3	44.8	4.7	49.1	5.3	41.5	3.8
Group 4	27.7	9.1	29.0	9.4	26.7	8.8

Unit: pixels.

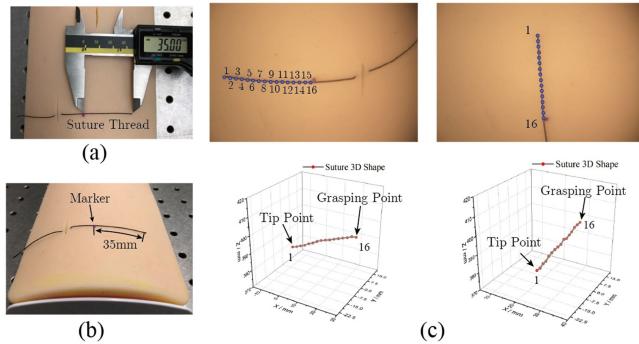


Fig. 12. (a) and (b) Suture thread lies on a tilt artificial tissue's surface. (c) 3-D coordinates computation of suture thread's different orientations.

C. Experimental Results of 3-D Coordinates Computation of Suture Thread

A suture thread was attached to an artificial tissue with a declining surface in order to introduce a height deviation between the tip and its grasping point, which was shown in Fig. 12(a) and (b).

To facilitate the measurement for comparison, the distance between the suture thread's tip and grasping point was set to 35 mm in advance, and the tissue was moved to four different locations with respect to the camera frame. Substituting the detected key point correspondences into the *triangulation* relationship, the 3-D spatial coordinates can be obtained. The suture thread's shape and total length can be achieved by connecting these 3-D points together, and two results were illustrated in

TABLE IV
COMPUTATIONAL RESULTS OF THE SUTURE THREAD'S LENGTH BETWEEN THE TIP AND GRASPING POINTS WITH DIFFERENT ORIENTATIONS

Orientation	1	2	3	4
Computed length	35.8	35.15	33.8	33.4
Length error	0.8	0.15	1.2	1.6

Unit: mm.

Fig. 12(c), and computational outcomes of four experiments were presented in Table IV.

The constructed geometry showed a straight line with height deviation at two ends, and the maximum length error was reported as 1.6 mm, which was less than the 5% error percentage of the suture thread's total length. Given that the computed lengths and the 3-D shape both matched well with the ground truths, the feasibility and accuracy of the proposed model-free approach were consequently proved.

D. Robustness Tests and Analysis of 3-D Pose Computation

To comprehensively evaluate the method, a suture was pierced manually and configurated with three different lengths between the cutting tip and the grasping point, which were 15, 25, and 40 mm long, respectively. A grained cloth was selected as the background, and a blob (marker) was attached to the suture thread to mark the grasping point.

In each group, six experiments were conducted using arbitrarily orientated suture threads, and typical results were shown in Fig. 13(a). Besides, the robustness was further examined by changing the background to two different artificial tissues, and the computational results were presented in Fig. 13(b). To validate the performance of the 3-D reconstruction for a suture thread with practical length, a long thread with a 132 mm length was also adopted to construct quasi circles and heart shapes from its exist point to the cutting tip. Setting the grasping point at the exit point, its complete shape can be built up. For each specific shape, the suture thread was placed at different locations, and six experiments were carried out. Three typical results were shown in Fig. 13(c) and (d). The outcomes of all experiments were summarized in Table V. The overall computational time of each

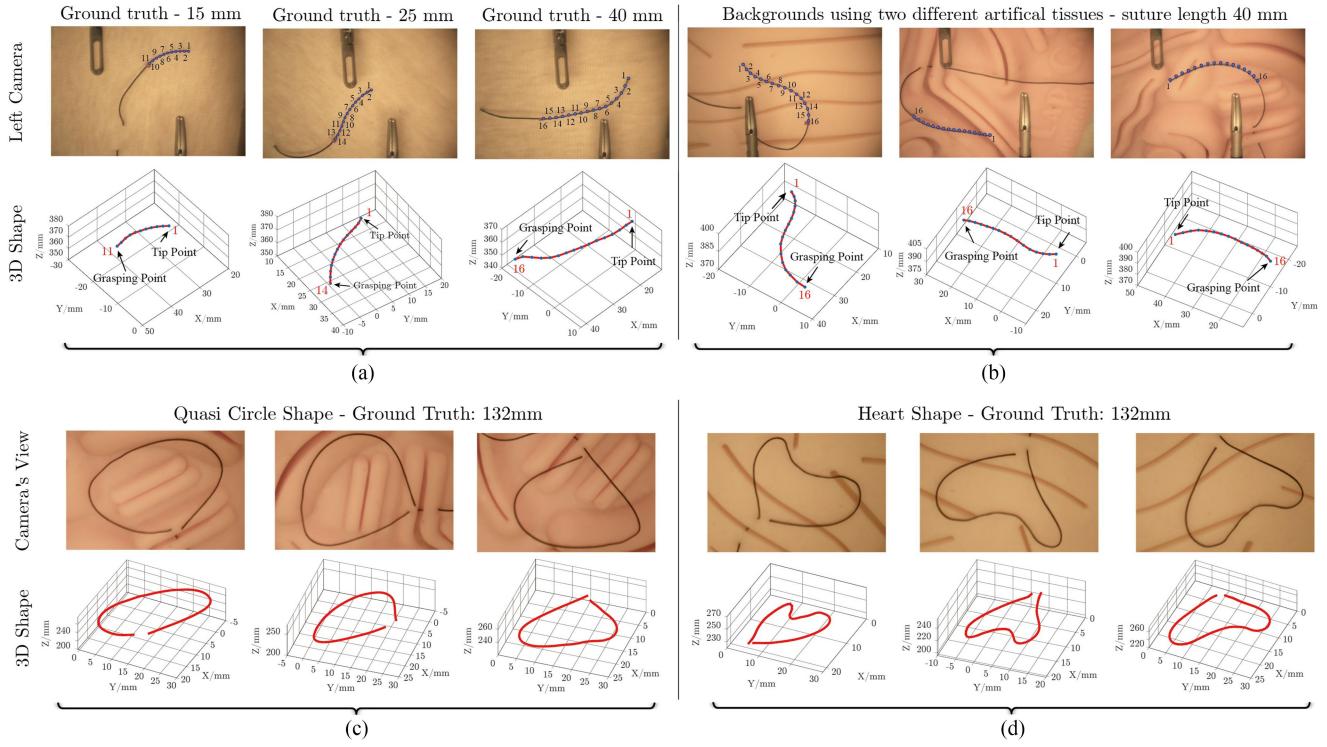


Fig. 13. Computed 3-D shape of the suture thread with different lengths between the tip and grasping point on various backgrounds. (a) Different lengths of a suture thread on a grained cloth. (b) 3-D coordinates computation on two artificial tissues. (c) Long suture thread's 3-D shape computation with specific shapes on artificial tissues.

TABLE V
COMPUTED LENGTHS OF THE SUTURE THREAD WITH DIFFERENT LENGTHS AND SHAPES

Material	Ground Truths	Group 1	Group 2	Group 3	Group 4	Group 5	Group 6	Average error	Maximum error
Grained Cloth	15	16.2	15.9	13.5	14.5	16.6	16.7	1.23	1.7
	25	24.3	25.8	25.1	25.2	26.1	24.5	0.57	1.1
	40	39.8	42.8	38.5	39.8	39.9	41.4	1.03	2.8
Artificial Tissue 1	40	39.6	38.2	41.9	41.9	41.6	41.8	1.57	1.9
	132(Quasi Round)	128.5	138.6	137.9	129.2	135.9	132.4	4.02	6.6
Artificial Tissue 2	40	40.9	40.8	40.02	39.5	39.5	40.9	0.60	0.9
Artificial Tissue 2									
132(Heart Shape)									

Unit: mm.

experiment varies from 40 to 60 s due to different environmental noises.

As can be seen in the figure, a smooth and curved line between suture thread's tip and grasping point was successfully generated in each group, and each 3-D shape closely matched its prospective 2-D views without singular points. Referring to the computed lengths, the average errors of each group were reported as 1.23, 0.57, 1.03, 1.57, 4.52, 0.60, and 3.52 mm. Large errors also existed during random tests, and it is noticed 6.6 and 5.0 mm errors were reported when computing the 3-D length of the longest suture thread.

Compared to [45], they put their main efforts on the 2-D segmentation of a suture thread based on a learning method, and its 3-D coordinates were not computed in their studies. Besides, our method eliminated the time-consuming data collection and predictor's training before carrying out this suture thread's segmentation and its shape computation task. When

evaluating average error percentages of long suture threads that were calculated as 3.04% and 2.29% using our method, a better performance was obtained compared with the method presented in [29]. Considering our model-free method rather than a model-based approach in [29], and our results were obtained from more complex operating environments, it can be concluded the approach reported in this paper can be more suitable and reliable for a suture thread's 3-D computation in various scenarios.

During the manipulation, a long suture thread might be messed up together, such as self-intersection condition that was shown in Fig. 14. Under these circumstances, our method is also feasible to compute its 3-D shape, as well as the desired grasping point. With a segmented suture thread, the algorithm picks out key points along its contour in every η_2 until reaching the desired grasping point, and then it starts from the tip again and figures out other key points along the remaining contour

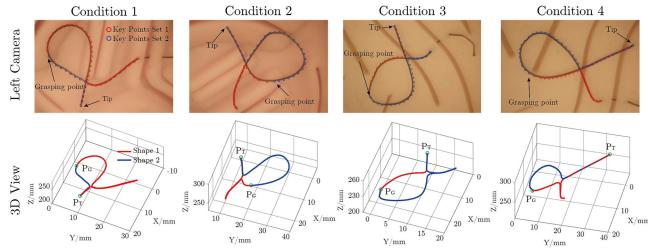


Fig. 14. 3-D reconstruction of a self-intersected suture thread.

until traverses all contour points. These key points were divided into two sets, and they were highlighted using different colors. Accordingly, their stereo correspondences can be located from the right camera's frame, and the 3-D shape and the desired grasping point P_G of a crossed suture thread can be thereby achieved.

V. CONCLUSION

This paper proposed a novel IPdIC algorithm to help surgeons evaluate the 3-D position of a conventional suture thread with no distinctive features or markers through a stereo camera system. After clicking the suture thread's tip P_T and its grasping P_G in the left camera, a proposed CCIC algorithm can thin the image into a contour pattern following a three-layer workflow, and accurately locate these two points regardless of manual click errors and environmental noises. Then, key points along the suture thread can be autonomously generated in the left camera.

To solve the problem due to the suture thread in an arbitrary shape with unknown depth information, a maximum virtual triangle based iterative computation was adopted for 3-D reconstruction. By updating the geometry relationship iteratively, stereo correspondences of the planned key points can be obtained in the right camera. Refinement algorithms were also employed for each newly obtained point to enhance its accuracy. Experiments associated with the tip and grasping point detections in two cameras and the 3-D reconstruction of the suture thread in various conditions were conducted. Special cases such as sudden drop and self-intersection of a suture thread were also examined. The results confirm that key point correspondences in two cameras can be accurately evaluated at the pixel level and provide accurate information for 3-D shape computation, proving the performance and robustness of the proposed scheme.

REFERENCES

- [1] R. H. Taylor, A. Menciassi, G. Fichtinger, P. Fiorini, and P. Dario, "Medical robotics and computer-integrated surgery," *IEEE Trans. Robot. Autom.*, vol. 19, no. 5, pp. 765–781, Oct. 2003.
- [2] E. P. Westerbeek-van der Putten, R. H. Goossens, J. J. Jakimowicz, and J. Dankelman, "Haptics in minimally invasive surgery—A review," *Minimally Invasive Therapy Allied Technol.*, vol. 17, no. 3, pp. 3–16, 2008.
- [3] Y. Dai, Y. Xue, and J. Zhang, "Bio-inspired integration of auditory and haptic perception in bone milling surgery," *IEEE/ASME Trans. Mechatron.*, vol. 23, no. 2, pp. 614–623, Apr. 2018.
- [4] M. J. Mack, "Minimally invasive and robotic surgery," *J. Amer. Med. Assoc.*, vol. 285, pp. 568–572, 2001.
- [5] V. Karri, L. R. Chang, Y. T. Liu, and H. C. Chen, "'Through-the-loop' tie for microsurgical suturing," *J. Plastic, Reconstructive Aesthetic Surg.*, vol. 63, no. 7, pp. 1087–1090, 2010.
- [6] O. F. Duenas-Garcia, G. M. Sullivan, K. Leung, K. L. Billiar, and M. K. Flynn, "Knot integrity using different suture types and different knot-tying techniques for reconstructive pelvic floor procedures," *Int. Urogynecol. J.*, vol. 29, no. 7, pp. 979–985, 2018.
- [7] K. Hyosig and J. T. Wen, "Robotic knot tying in minimally invasive surgeries," in *Proc. Int. Conf. Intell. Robots Syst.*, 2002, pp. 1421–1426.
- [8] H. Kang and J. T. Wen, "Autonomous suturing using minimally invasive surgical robots," in *Proc. IEEE Int. Conf. Control Appl.*, 2000, pp. 742–747.
- [9] R. Jackson and M. Cavusoglu, "Needle path planning for autonomous robotic surgical suturing," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2013, pp. 1669–1675.
- [10] S. Sen, A. Garg, D. V. Gealy, S. McKinley, Y. Jen, and K. Goldberg, "Automating multi-throw multilateral surgical suturing with a mechanical needle guide and sequential convex optimization," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2016, pp. 4178–4185.
- [11] D. L. Chow and W. Newman, "Improved knot-tying methods for autonomous robot surgery," in *Proc. IEEE Int. Conf. Autom. Sci. Eng.*, 2013, pp. 461–465.
- [12] D. L. Chow and W. Newman, "Trajectory optimization of robotic suturing," in *Proc. IEEE Int. Conf.*, 2015, pp. 1–6, 2015.
- [13] B. Lu, H. K. Chu, and L. Cheng, "Robotic knot tying through a spatial trajectory with a visual servoing system," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2017, pp. 5710–5716.
- [14] B. Lu, H. K. Chu, K. C. Huang, and L. Cheng, "Vision-based surgical suture looping through trajectory planning for wound suturing," *IEEE Trans. Autom. Sci. Eng.*, vol. 16, no. 2, pp. 542–556, Apr. 2019.
- [15] E. K. Batra *et al.*, "Influence of emergency physician's tying technique on knot security," *J. Emergency Med.*, vol. 10, no. 3, pp. 309–316, 1992.
- [16] D. Lowe, "Object recognition from local scale-invariant features," in *Proc. IEEE Int. Conf. Comput. Vis.*, 1999, pp. 1150–1157.
- [17] D. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, pp. 91–110, 2004.
- [18] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2005, pp. 886–893.
- [19] H. Bay, A. Ess, T. Tuytelaars, and L. V. Gool, "Speeded up robust features (SURF)," *Comput. Vis. Image Understanding*, vol. 10, pp. 346–359, 2008.
- [20] A. Mavrinac, X. Chen, and J. L. Alarcon-Herrera, "Semiautomatic model-based view planning for active triangulation 3-D inspection systems," *IEEE/ASME Trans. Mechatron.*, vol. 20, no. 2, pp. 799–811, Apr. 2015.
- [21] H. Wang, B. Yang, J. Wang, X. Liang, W. Chen, and Y. Liu, "Adaptive visual servoing of contour features," *IEEE/ASME Trans. Mechatron.*, vol. 23, no. 2, pp. 811–822, Apr. 2018.
- [22] Z. Li, T. Zhao, F. Chen, Y. Hu, C. Su, and T. Fukuda, "Reinforcement Learning of Manipulation and Grasping using Dynamical Movement Primitives for a Humanoid-like Mobile Manipulator," *IEEE/ASME Trans. Mechatron.*, vol. 23, no. 1, pp. 121–131, Feb. 2018.
- [23] C. Becker *et al.*, "Supervised feature learning for curvilinear structure segmentation," in *Proc. Med. Image Comput. Comput. Assisted Interventions Conf.*, 2013, pp. 526–533.
- [24] S. Tsogkas and I. Kokkinos, "Learning-based symmetry detection in natural images," in *Proc. Eur. Conf. Comput. Vis.*, 2012, pp. 41–54.
- [25] B. D. Argall, S. Chernova, M. Veloso, and B. Browning, "A survey of robot learning from demonstration," *Robot. Auton. Syst.*, vol. 57, no. 5, pp. 469–483, 2009.
- [26] S. Javadani *et al.*, "Shared autonomy via hindsight optimization for teleoperation and teaming," *Int. J. Robot. Res.*, vol. 37, no. 7, pp. 717–742, 2018.
- [27] N. Padoy and G. D. Hager, "Deformable tracking of textured curvilinear objects," in *Proc. Brit. Mach. Vis. Conf.*, 2012, pp. 5.1–5.11.
- [28] T. Liu and M. C. Cavusoglu, "Needle grasp and entry port selection for automatic execution of suturing tasks in robotic minimally invasive surgery," *IEEE Trans. Autom. Sci. Eng.*, vol. 13, no. 2, pp. 552–563, Apr. 2016.
- [29] R. C. Jackson, R. Yuan, D.-L. Chow, W. S. Newman, and M. C. Cavusoglu, "Real-time visual tracking of dynamic surgical suture threads," *IEEE Trans. Autom. Sci. Eng.*, vol. 15, no. 3, pp. 10798–1090, Jul. 2018.
- [30] W. K. Wong, B. Yang, C. Liu, and P. Poignet, "A quasi-spherical triangle-based approach for efficient 3-D soft-tissue motion tracking," *IEEE/ASME Trans. Mechatron.*, vol. 18, no. 5, pp. 1472–1484, Oct. 2013.
- [31] H. Mayer, I. Nagy, A. Knoll, E. U. Schirmbeck, and R. Bauernschmitt, "The Endo[PA]R system for minimally invasive robotic surgery," in *Proc. Int. Conf. Intell. Robots Syst.*, 2004, pp. 3637–3642.

- [32] M. Simi *et al.*, "Magnetically activated stereoscopic vision system for laparoendoscopic single-site surgery," *IEEE/ASME Trans. Mechatron.*, vol. 18, no. 3, pp. 1140–1151, Jun. 2013.
- [33] L. Lu, Y. Ding, Y. Luan, Y. Yin, Q. Liu, and J. Xi, "Automated approach for the surface profile measurement of moving objects based on PSP," *Opt. Express*, vol. 25, no. 25, pp. 32120–32131, 2017.
- [34] S. Barone *et al.*, "Three-dimensional point cloud alignment detecting fiducial markers by structured light stereo imaging," *Mach. Vis. Appl.*, vol. 23, no. 2, pp. 217–229, 2012.
- [35] N. Kanopoulos, N. Vasanthavada, and R. L. Baker, "Design of an image edge detection filter using the Sobel operator," *IEEE J. Solid State Circuits*, vol. 23, no. 2, pp. 358–367, Apr. 1988.
- [36] J. Canny, "A computational approach to edge detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 8, no. 6, pp. 679–698, Nov. 1986.
- [37] B. Paul, L. Zhang, and X. Wu, "Canny edge detection enhancement by scale multiplication," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 9, pp. 1485–1490, Sep. 2005.
- [38] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge, U.K.: Cambridge Univ. Press, 2003.
- [39] J. L. Rodgers and W. A. Nicewander, "Thirteen ways to look at the correlation coefficient," *Amer. Statistician*, vol. 42, no. 1, pp. 59–66, 1988.
- [40] Z. Zhang, "A flexible new technique for camera calibration," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 11, pp. 1330–1334, Nov. 2000.
- [41] J. Heikkilä and O. Silven, "A four-step camera calibration procedure with implicit image correction," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 1997, pp. 1106–1112.
- [42] K. Nomizu and T. Sasaki, *Affine Differential Geometry*. Cambridge, U.K.: Cambridge Univ. Press, 1994.
- [43] P. Torr and A. Zisserman, "MLESAC: A new robust estimator with application to estimating image geometry," *Comput. Vis. Image Understanding*, vol. 78, pp. 138–156, 2000.
- [44] R. I. Hartley and P. Sturm, "Triangulation," *Comput. Vis. Image Understanding*, vol. 68, no. 2, pp. 146–157, 1997.
- [45] Y. Hu, Y. Gu, J. Yang, G. Yang, "Multi-stage suture detection for robot assisted anastomosis based on deep learning," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2018, pp. 4826–4833.



Bo Lu received the B.Eng. degree in ship and offshore engineering from the Dalian University of Technology, Dalian, China, in 2013, and the M.S. (Hon.) and Ph.D. degrees in mechanical engineering from The Hong Kong Polytechnic University, Hong Kong, in 2015 and 2019, respectively. He is currently a Postdoctoral Research Fellow with the T-stone Robotics Institute, The Chinese University of Hong Kong, Hong Kong. His current research interests include medical robotics, computer vision, vision-based manipulation, automation and control, trajectory planning, and surgical tool detection.



Henry K. Chu (M'12) received the B.S. degree in mechanical engineering from the University of Waterloo, Waterloo, ON, Canada, and the M.S. and Ph.D. degrees in mechanical and industrial engineering from the University of Toronto, Toronto, ON, Canada.

He was a Postdoctoral Fellow with the University of Toronto and the City University of Hong Kong, Hong Kong. He is currently an Assistant Professor with The Hong Kong Polytechnic University, Hong Kong. His research interests include robotic manipulation, vision-based control and automation, microsystem design, and tissue engineering.



Kaicheng Huang received the B.Eng. degree in automation from the Department of Mechatronic and Control Engineering, Shenzhen University, Shenzhen, China, in 2014, and the M.Sc. degree in mechanical and automation engineering from The Chinese University of Hong Kong, Hong Kong, in 2015. He is currently working toward the Ph.D. degree in mechanical engineering with The Hong Kong Polytechnic University, Hong Kong.

His research interests include automated cell patterning with dielectrophoresis.



Jiewen Lai received the B.Eng. degree in metallurgical engineering from the Wuhan University of Science and Technology, Wuhan, China, in 2016, and the M.Sc. degree in mechanical and automation engineering from The Chinese University of Hong Kong, Hong Kong, in 2017. He is currently working toward the Ph.D. degree in mechanical engineering with The Hong Kong Polytechnic University, Hong Kong.

His research interests include continuum robot, soft robot, robotic control, computer vision, and surgical robot system.