Journal of Medical Robotics Research, Vol. 0, No. 0 (2013) 1–13

Development and Evaluation of a Markerless 6 DOF Pose Tracking Method for a Suture Needle from a Robotic Endoscope

Yiwei Jiang[a], Haoying Zhou[a], Gregory S. Fischer[a]

[a] *Department of Robotics Engineering, Worcester Polytechnic Institute, 100 Institute Rd, Worcester, Massachusetts 01609, USA*
E-mail: yjiang5@wpi.edu

Remarkable progress has been made in the field of robot-assisted surgery in recent years, particularly in the area of surgical task automation, though many challenges and opportunities still exist. Among these topics, the detection and tracking of surgical tools play a pivotal role in enabling autonomous systems to plan and execute procedures effectively. For instance, accurate estimation of a needle's position and posture is essential for surgical systems to grasp the needle and perform suturing tasks autonomously. In this paper, we developed image-based methods for markerless 6 degrees of freedom (DOF) suture needle pose estimation using keypoint detection technique based on Deep Learning and Point-to-point Registration, we also leveraged multi-viewpoint from a robotic endoscope to enhance the accuracy. The data collection and annotation process was automated by utilizing a simulated environment, enabling us to create a dataset with 3446 evenly distributed needle samples across a suturing phantom space for training and to demonstrate more convincing and unbiased performance results. We also investigated the impact of training set size on the keypoint detection accuracy. Our implemented pipeline that takes a single RGB image achieved a median position error of 1.4 mm and a median orientation error of 2.9 degrees, while our multi-viewpoint method was able to further reduce the random errors.

Keywords: Surgical Robotics; Markerless Tracking; Keypoint Detection; Deep Learning.

1. Introduction

In Minimally Invasive Surgeries (MIS), when surgeons manipulate tools to perform suturing instead of using their hands, the suturing procedure becomes more demanding and tedious, while it still requires high dexterity. The success of the surgery and the patient's well-being is directly influenced by the quality of suturing and the time taken to complete the procedure. Therefore, it is important to reduce the physical and mental burden of the surgeon in this situation. As a result, there has been a growing interest in automating suturing through robotic assistance, and advanced suturing is a technique that can be expected to reach Level 3 - Conditional Autonomy.[1] To automate suturing with a robot, a crucial prerequisite is that the robot needs to know where the needle locates and to track the pose so that the robot can pick up a needle and adjust its suturing movements according to the latest estimated needle pose. Nevertheless, tracking a small needle efficiently, especially in dynamic surgical environments with varying lighting conditions and complex tissue backgrounds, remains a challenging problem.

Besides optical tracking products like Aurora (Northern Digital Inc., Canada), many researchers favor pattern-based markers, such as ArUco markers,[2] because they only require an image from a typical camera to track, which is readily available or easily integratable on many existing robotic surgical systems. Despite the reduced hardware restrictions, these markers can still achieve good pose estimation accuracy of less than 0.7 mm translation error and 0.85 deg rotation error.[3] The flexibility of these markers allows their application in diverse scenarios. For instance, Qian et al. used the integrated camera on a head-mount display to track such a marker and determine the pose of the surgeon's head relative to the surgical robot.[4] Also, these markers can also be scaled as small as 6 mm x 6 mm, to be attached to a laparoscopic photoacoustic probe for image reconstruction.[5] However, for some other surgical instruments, such as the forceps in the da Vinci surgical system (Intuitive Surgical Inc, USA), a markerless tracking approach is the preferred choice,[6] and this preference extends to suture needles tracking as well.

To address the challenges mentioned above, we proposed a markerless method based on Deep Learning and Point-to-point Registration that only uses RGB images to estimate the 6 DOF pose of a suture needle[7] under a pub-

lic, standard surgical scene simulator[8] (Fig. 1). Our method does not require any modification to a commercial suture needle and introduces little interference to the existing surgical workflows.

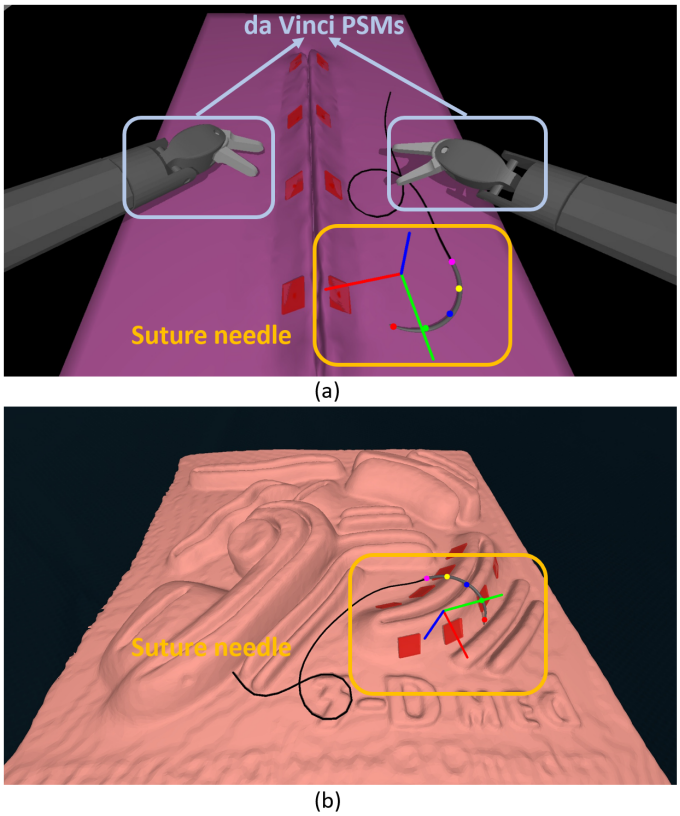

Fig. 1. Visualization of the needle pose tracking results given the input of images captured from a robotic endoscope. (a) Suturing training environment provided by the 2021-2022 AccelNet Surgical Robotics Challenge, PSM refers to Patient Side Manipulator of the da Vinci surgical system. (b) Scanned 3-Dmed suturing pad provided by the 2023-2024 AccelNet Surgical Robotics Challenge.

In this paper, we made several improvements to further extend our previous work. Firstly, the dataset collection and annotation process was automated, this allowed us to generate a significantly larger dataset of 3446 samples, and the needle positions were evenly distributed across the phantom space. This expanded dataset served for training and enabled us to demonstrate more convincing and unbiased performance evaluation results. The impact of set size on the keypoint detection accuracy was investigated by experimenting with the model using different sizes of training sets. Secondly, a new suture needle model with a different size and shape was created, and a more realistic suture pad was included as a background. This demonstrates the extensibility of our approach. Furthermore, we conducted a more comprehensive analysis. We adopted relative L2 error with respect to the needle bounding box to assess the keypoint localization accuracy more informatively. We also included a thorough examination of the sample error distribution and tested the extensibility of our approach. This analysis offers insights into the strengths and limitations of our approach, providing a deeper understanding of its performance and effectiveness.

2. Related Work

Over the years, a variety of approaches have been developed to track suture needles and estimate their poses, evolving from color-based detection to advanced computer vision techniques. Early efforts, such as those by Wengert et al.,[9] Kurose et al.,[10] and Sen et al.,[11] although achieved sub-millimeter position accuracy and less than 3° orientation accuracy, their systems relied on painting and the help of optimal lighting conditions or camera settings. As evidenced by a 73% needle contour detection rate reported in [12] due to specular highlights and so on. In [13], three green markers were put on the needle to ensure precise detection, but attaching physical markers would make the needle not suitable for many suturing tasks. These color-based approaches are usually limited by the modifications to the suture needles and the need for environment-specific tuning of parameters, which suffers to accommodate the complex and varying surgical background with different objects, changed light conditions, or various tissue textures.

More recently, advanced computer vision techniques including Deep Learning have been explored to address some of the limitations. Ferro et al.[14] introduced Bayesian filters with diverse observation models to account for uncertainties in needle features and motion. Mei et al.[15] utilized two popular object detection architectures: YOLO (You Only Look Once)[16] and R-CNN (Region-based Convolutional Neural Network)[17] to extract the bounding box of a suture needle in the images. Zhou et al.[18] also used Feature Pyramid Net (FPN) to detect a tiny needle tip with an accuracy of 0.55 (Intersection over Union) in the confidence of 99.2%. While these approaches excelled in 2D object detection/segmentation, they lacked in providing 6 DOF pose information. Wilcox et al.[19] combined semantic segmentation with random sample consensus (RANSAC) to obtain an estimated needle pose but did not include a numerical evaluation of the accuracy. In a separate line of research, studies in [20] and later in [21] focused on the needle circular geometries and their elliptical projections from extracted feature points, reporting error levels of 0.87 mm and 0.12 degrees in their simulated environments. Other researchers also incorporated physical constraints related to manipulator dynamics to enhance tracking robustness.[22] However, these tracking methods were designed to track continuous needle motion but not to estimate the needle pose from a single image. Lastly, an end-to-end pose estimation neural network model GDRnet[23] showed promising

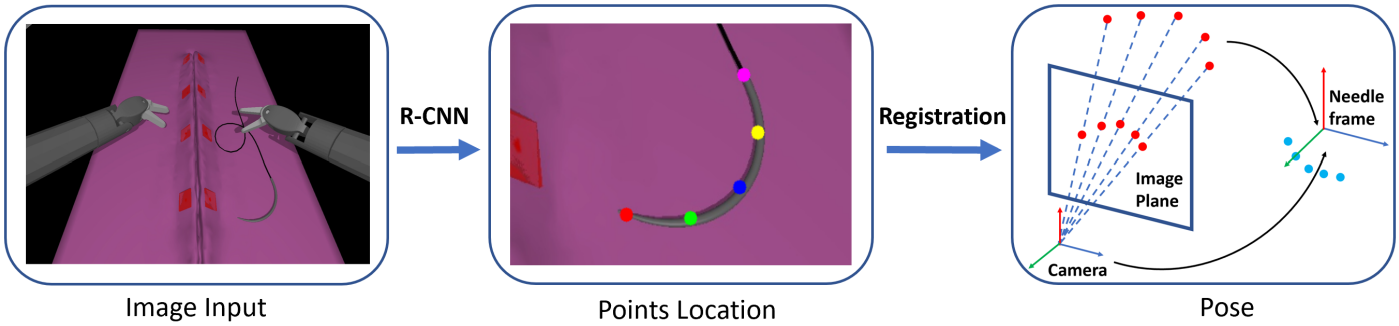

Fig. 2. Overview of the two-stage pose estimation method. We utilize R-CNN to localize pre-defined keypoints on the needle based on the image input, then calculate the needle pose using point set registration techniques.

center-meter level accuracy with various objects in a much larger spatial scale and may have the potential for better suture needle tracking performance with specific fine-tuning.

3. Methods

Our study aims to estimate the 6 DOF pose of a suture needle using only 2D RGB images from a robotics endoscope, without the need for markers or any modification to the off-the-shelf suture needle. The proposed method comprises two main steps. Firstly, we utilize a neural network architecture from Mask R-CNN,[24] which was trained on a customized dataset, to extract the pre-defined key points on the needle body from a 2D image. Then, we calculate the needle transformation with respect to the endoscope using point-to-point registration techniques using the 2D or 3D positional information and the correspondence of these needle key points. (Fig. 2)

Additionally, we leverage the robotic endoscope's ability to provide multiple viewpoints of the needle, which contributes to error reduction in our approach.

3.1. *Simulated Suturing Environment*

The simulated surgical scene used in this paper is a standard robotic suturing environment provided by 2021-2022 and 2023-2024 AccelNet Surgical Robotics Challenge,[8] and it is built based on Robot Operating System (ROS)[25] and Asynchronous Multi-Body Framework (AMBF).[26] The scene includes suturing training phantoms of two types: a simple virtually constructed phantom (Fig. 1.a), or a scanned realistic 3-Dmed Suture Pad (Fig. 1.b). On the phantoms, there are red squares representing the entry and exit holes for guiding the needle's path. The needle itself has a thread connected to the tail. Moreover, this setup is equipped with two da Vinci patient-side manipulators (PSMs) and one Endoscopic Camera Manipulator (ECM) from the da Vinci Research Kit (dVRK).[27] The images in Fig. 1 were captured from the left camera of the stereo endoscope on the ECM.

The needle model provided by AccelNet Surgical Robotics Challenge (Fig. 3 (a)) is essentially a 120-degree arc with a radius of 10.18 mm, we defined five body points (A, B, ..., E) which are equally spaced on the needle body. Note the coordinates of one of these points in the needle frame as $P_N = [x_N, y_N, z_N]$. To demostrate the extendibility of our method, we made another 150-degree suture needle with a radius of 9mm (Fig. 3 (b)).

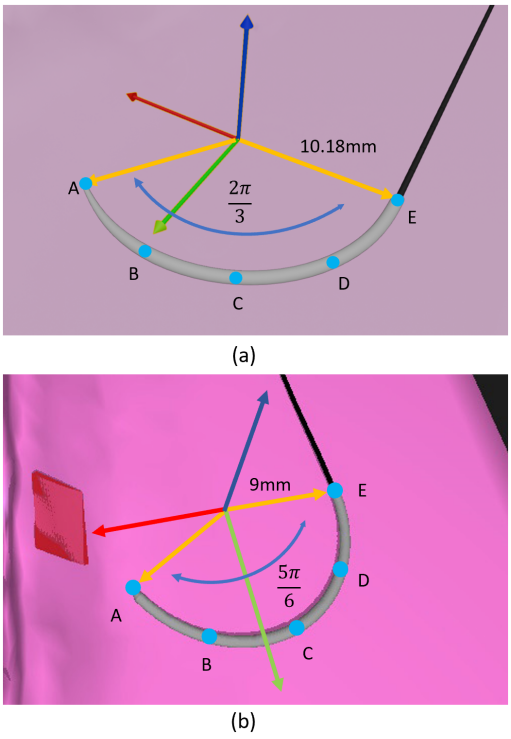

Fig. 3. Needle frame and keypoints definition. Five keypoints are evenly distributed over the needle body, Point A is the head of the needle, and Point E is the tail connected with a thread.

3.2. *Automatic Data Collection and Labeling*

Creating a dataset for training a keypoint detection model typically requires a substantial effort, involving image collection of the objects in various poses, followed by extensive manual annotation of all the bounding boxes and keypoints on the images as ground truth. Particularly in our case, wherein identifying the 3 middle keypoints on the needle precisely is challenging for humans as no salient features are associated with them.

Leveraging the benefits of a simulated environment, we streamlined and automated these processes. Regarding sample collection, by programmatically moving the object model to various locations and setting it to random poses in the simulator, we were able to collect thousands of unbiased image samples with random suture needle poses. Specifically, we divided a cuboid zone into grids and positioned the needle in the 360 corner locations, as elaborated in Section 4.1. As for annotation, in our simulator, the 6 DOF transformation between any two objects can be directly queried. With the 3D transformation from the camera coordinate system to the needle coordinate system, we can project any points on the needle body, including all the 5 defined key points, to the 2D images captured by the camera. (Eq. 1)

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \tag{1}$$

where x, y, z are 3D coordinates of a needle keypoint; u, v are projected 2D coordinates of that keypoint on the image; R, T are the rotation matrix and translation matrix of the camera; camera intrinsic parameters are:

$$\begin{gathered} f_x = f_y = \frac{H}{2tan(\frac{f_{va}}{2})}, c_x = \frac{W}{2}, c_y = \frac{H}{2} \\ f_{va} = 1.2, W = 1920, H = 1080. \end{gathered} \tag{2}$$

Obtaining the coordinates of the bounding box is then straightforward. The minimum values of all key points along each axis determine the coordinates of the upper left corner, while the maximum values dictate those of the lower right corner. Eq. (3).

$$\begin{aligned} (u_1, v_1) &= (\min_{\forall i \in needle} u_i, \min_{\forall i \in needle} v_i) \\ (u_2, v_2) &= (\max_{\forall i \in needle} u_i, \max_{\forall i \in needle} v_i) \end{aligned} \tag{3}$$

where i is any keypoint on the needle, u, v are the image coordinates of the keypoints.

Our automated approach facilitates the data collection process and eliminates the necessity for manual labeling and ensures accurate annotations. As a result, we were able to generate a large dataset of 6892 images within a mere 3-hour timeframe to be ready for use in training and testing with minimal effort.

3.3. *Keypoint Localization using R-CNN*

In recent years, the object detection area has witnessed the development of numerous approaches, primarily falling under two families, YOLO[16] and R-CNN.[17] Given that our objective is not confined to obtaining the bounding box of the needle but also the "landmarks" on the needle body. To accomplish this goal, we opted to use Keypoint R-CNN. Keypoint R-CNN is an extension of Mask R-CNN for keypoint detection.[24] It begins with a backbone network to extract features, a Region Proposal Network for candidate region selection, and uses ROI Align for precise feature extraction within these regions. Instead of the mask head in Mask R-CNN, which generates pixel-level masks for each object, Keypoint R-CNN uses the keypoint head that models a keypoint's location as a one-hot m^2 binary mask, where only a single pixel is labeled as foreground. We implemented a Keypoint R-CNN model with a pre-trained ResNet-50-FPN[28] backbone using the PyTorch library.[29] Fine-tuned the model on a customized dataset. The loss function contains three terms, which are the classification (Eq. 5) and regression losses (Eq. 7) for both the Region Proposal Network and the R-CNN,[30] and the keypoint loss in a form of cross-entropy loss over an m^2-way softmax output, m is the side length in pixels of the binary mask represents the training target, Eq. 6.

$$Loss = L_{cls} + L_{reg} + L_{keypts} \tag{4}$$

$$L_{cls} = -\frac{1}{N_{cls}} \sum_{i=1}^{N_{cls}} (y_{c,i} log \hat{y}_{c,i} + (1 - y_{c,i}) log(1 - \hat{y}_{c,i}))) \tag{5}$$

where N_{cls} is the number of classes, $y_{c,i}$ is 0 or 1 whether the region proposal i predicts class c is correct, $\hat{y}_{c,i}$ is the predicted probability.

$$L_{reg} = \lambda \frac{1}{N_{reg}} \sum_{i=1}^{N_{reg}} smooth_{L1}(t_i, t_i^*) \tag{6}$$

where λ is a hyperparameter, N_{reg} is the number of positive anchors, t_i represents the predicted bounding box, t_i^* represents the ground truth bounding box.

$$L_{keypts} = -\frac{1}{m^2} \sum_{1 \le i,j \le m} (y_{i,j} log \hat{y}_{ij}^k + (1 - y_{ij}) log(1 - \hat{y}_{ij}^k)) \tag{7}$$

where m is the side length of the binary mask, $y_{i,j}$ is 0 or 1 whether the prediction of pixel (i, j) is correct, $\hat{y}_{c,i}^k$ is the predicted probability that pixel (i, j) is keypoint k.

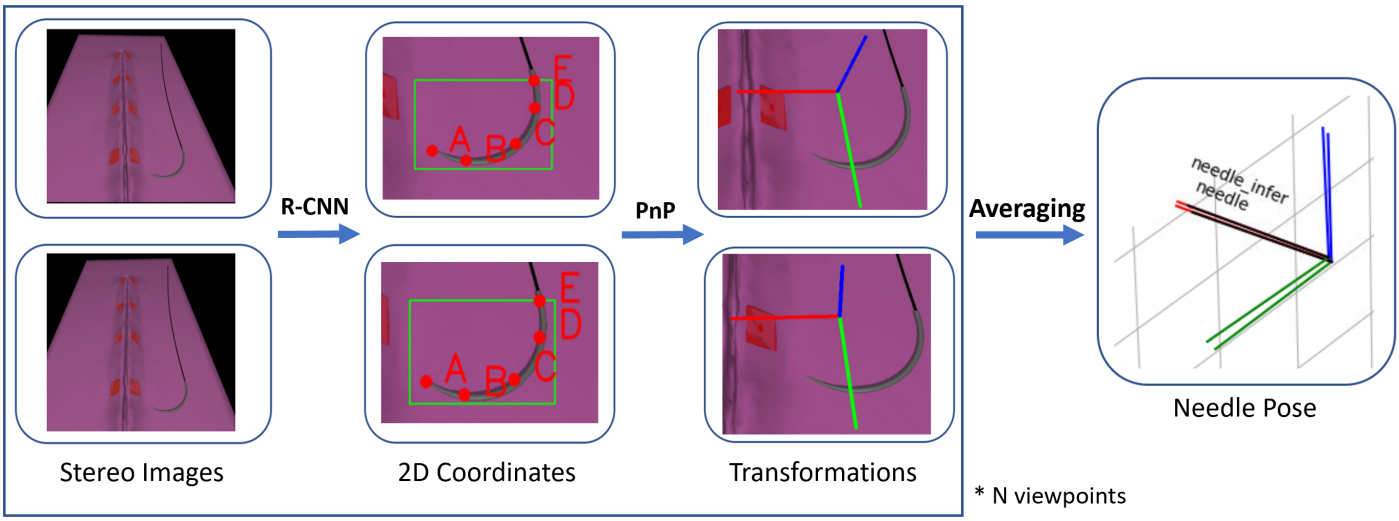

Fig. 4. Pipeline I. R-CNN extracts the defined 5 needle body points from the left and right images, the transformation from the left and right cameras to the needle are calculated respectively, the final result is an average of the two transformations. Note that the core algorithm of this pipeline does not require stereo images. Transformations from multiple viewpoints can be combined, see Section II. D.

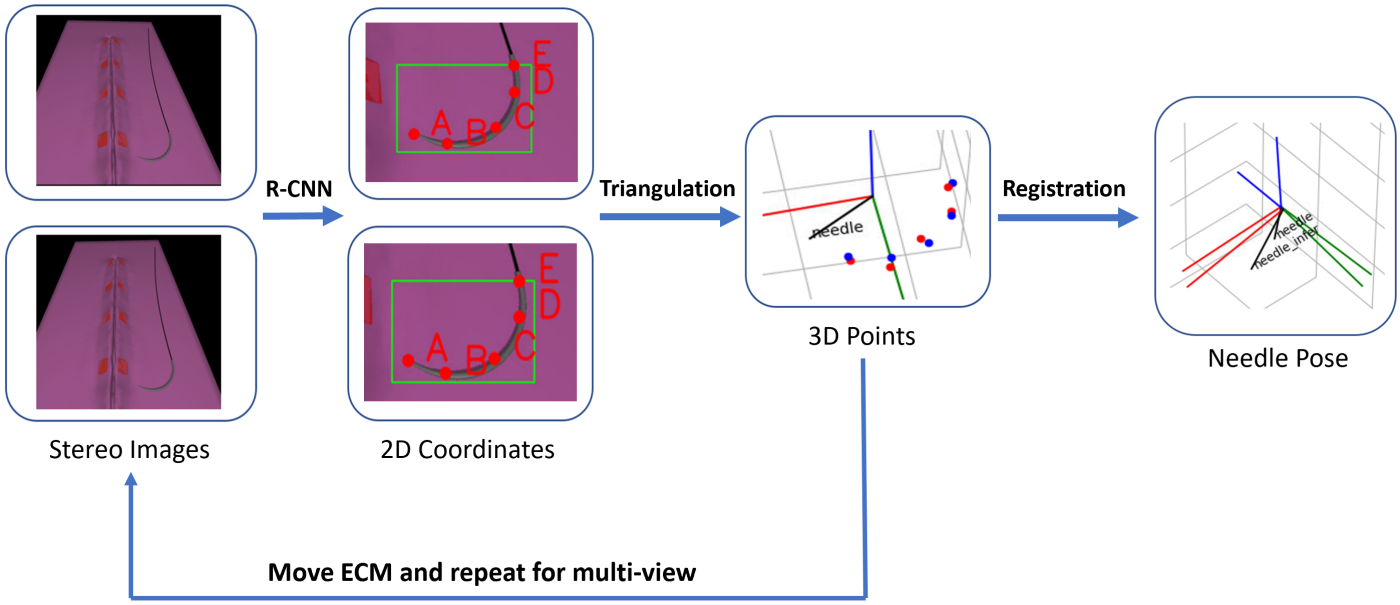

Fig. 5. Pipeline II. Both of the images are taken from the endoscope at the same time. 3D coordinates are triangulated from the 2D coordinates pairs (red dots are the ground truth, red ones are from stereo triangulation). The needle pose is the 3D point sets registration result.

3.4. *Point-to-point Registration*

Point-set registration is a technique to align two sets of points in space and can be used to determine the spatial transformation from a tracker to an object given the positional data from the tracker's and object's coordinate systems. In our specific case, we utilize this method to estimate the 6 DOF pose of a suture needle relative to a camera, based on the positional information of the needle's

keypoints in both the camera's and its own coordinate systems.

We implemented two distinct pipelines to achieve this goal. The first pipeline directly performs 2D-to-3D registration using the 2D coordinates of the key points on an image. In contrast, the second pipeline first determines a 3D point set using stereo triangulation, then registers the point set to the needle's frame and computes the pose of the needle.

Pipeline I - 2D-to-3D Registration

Based on the detected 2D image coordinates of the needle key points from the previous step, along with the corresponding 3D coordinates of these points in the needle's local frame, the task of estimating the pose of the needle with respect to the camera is referred to as the Perspective-n-Point[31] problem. To solve the PnP problem, we utilized the Efficient Perspective-n-Point (EPnP) method[32] to directly calculate the 6 DOF transformation. The complete pipeline for one pair of images is as Fig. 4 shows. The final output is an average of the two transformations. Averaging the translation part is straightforward. As for the rotation part, we converted the rotation metrics into quaternions, the two quaternions are averaged and then transformed back into a rotation matrix.

Pipeline II - 3D-to-3D Registration

An alternative method to register and compute the transformation is by using 3D point set coordinates. Utilizing the paired 2D coordinates of the key points on the left and right camera, along with the camera parameters, we can use stereo triangulation to calculate the 3D coordinates of these points. Finally, given the shape of the needle and the local 3D coordinates of these body points, we can perform a point cloud registration (PCR) with known correspondence to obtain the 6 DOF pose of the needle with respect to the camera. We used Direct Linear Transformation[33] (DLT) method to triangulate the key points. Arun's method[34] was used to calculate the homogeneous transformation from the point set in the camera frame to it in the needle frame, which is equivalent to the needle pose relative to the camera. The complete pipeline is as Fig. 5 shows.

3.5. *Multi-viewpoint from a Robotic Endoscope*

The ECM is a robotic arm with an endoscope, and the endoscope pose can be calculated from ECM's forward kinematics. When performing an MIS with the da Vinci robot, the surgeon can use the Master Tool Manipulator (MTM) to adjust the pose of the endoscope to get a different viewpoint. In suture needle tracking, self-occlusion is a special case when one part of the needle obstructs another part of it from a certain point of view so that the camera can not see its full body. In some particular viewpoints, the projection of the needle shape into a 2D image can even shrink to a line segment rather than a curve. To mitigate this problem and take advantage of the robotic endoscope, we introduce multi-viewpoint tracking to enhance our core algorithm. The surgeon can move the ECM to a few different poses and get multiple different views of the needle. Alternatively, a set of ECM poses surrounding the region of interest can be pre-selected or generated, so that the robot can automatically sweep these viewpoints. A valid image will be used only if the neural network reports a confidence score larger than a threshold, ensuring that the endoscope captures the entire needle and detects the keypoints for subsequent pose computation via EPnP.

The workflow is as Fig. 6 shows. Transformations estimated from valid image samples are stored in a queue, every time the ECM moves to a new pose, a new transformation inserts into the queue, and all the stored transformations are multiplied by the offset to convert them to the current endoscope pose. The tracking result is the average of all elements in the queue after eliminating outliers, which are identified based on Eq.8 and Eq. 9, to reduce random errors.

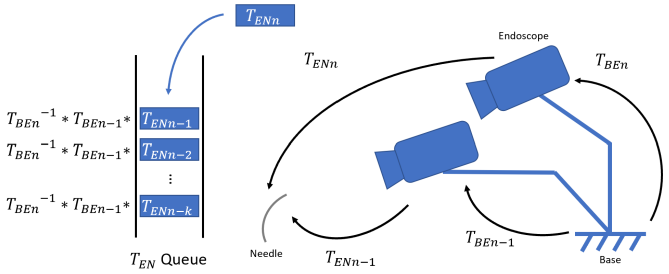

Fig. 6. ECM Multi-viewpoint ensemble. When the ECM moves from the previous pose (Index n-1) to the current pose (Index n), the latest estimation T_{ENn} is inserted into the queue, all previous results ($T_{ENn-1}, ..., T_{ENn-k}$, k is the queue size) are converted into the current endoscope frame for comparison and averaging.

$$p = (x, y, z, i, j, k) \tag{8}$$

6-dimensional pose vector representing the combination of translation and rotation vectors.

$$s(i) = |p(i) - mean(P_i)/std(P_i)| \tag{9}$$

Scores are computed axis-wise, P_i is the set of all i^{th} values in the pose vectors. If any of the 6 scores for a pose sample is beyond the set threshold, that sample is an outlier.

4. Experiments and Evaluation Metrics

4.1. *Single Viewpoint*

Within a cuboid zone of 50 mm (horizontal) × 10mm (vertical) × 90 mm (distance to the camera, 50 - 140 mm), we partitioned the space into gird cells of $5\,mm \times 5\,mm \times 5\,mm$, then move the needle to each of the grid corners sequentially. After each movement, the needle's orientation (roll, pitch, yaw) was randomly set 10 times. The image data and ground truth pose data were recorded corresponding to every distinct 6DOF pose.

This process resulted in a total of 3600 samples, encompassing various needle poses evenly distributed across the space. In total, 7200 labeled images were captured from both the left and the right cameras. Finally, 3446 samples were included in the dataset, 154 (4.3%) samples were excluded because of corrupted data or asynchronous alignment between image and pose data. The yield rate can be improved but is out of the scope of this project.

In addition to the main dataset, we collected samples of another 150-degree needle (Fig. 3(b)), and used the scanned 3-Dmed suturing training pad (Fig. 1(b)). The da Vinci PSMs were also included due to their color similarity to the needle. This augmentation allowed us to further evaluate the robustness and performance of our proposed methods in different scenarios where background interference is present.

4.2. *Multi-viewpoint*

In addition to the default ECM pose, we incorporated 10 alternative poses, thereby surrounding the region of interest from multiple perspectives. To evaluate the multi-view idea, we performed five auto-sweep tests, during which the ECM was moved to each of the 10 predefined poses sequentially, and the needle pose was estimated from all these viewpoints. Upon the completion of the sweep, outliers were identified and removed, allowing us to subsequently average the remaining inlier transformations as the final results. We also intentionally selected 3 failed cases of a single viewpoint, ECM to different poses, and eliminated the outliers. In each test, the needle was manually thrown on the phantom by a human, intentionally making the key points tend to be co-linear. Then the first three joints of the ECM were adjusted by small random angles within $(-\pi/10, \pi/10)$ for 2 times.

4.3. *Evaluation Metrics*

To assess the performance of our proposed methods, a few metrics are defined as follows.

Regarding 2D keypoint detection, we initially used the Euclidean distance (L2 norm error) between the model-predicted keypoint coordinates in pixels and ground truth, Eq. 10. This metric has a limitation as it does not consider variation in needle sizes appearing on the image because of the distances from the camera to the needle. For instance, a small value of an absolute L2 norm error in pixels indicates a smaller error when the needle is closer to the camera, while the same value may imply a larger localization error when the needle is far from the camera because the needle appears much smaller. To overcome this limitation, we propose relative L2 norm error as Eq. 11.

$$L2.\,2D.\,err_p = \sqrt{(u_p - u'_p)^2 + (v_p - v'_p)^2} \tag{10}$$

where p is an arbitrary keypoint, u_p, v_p are ground truth images coordinates of a keypoint, u'_p, v'_p are model estimated coordinates.

$$rel.\,L2.\,2D.\,err_{(i,p)} = \sqrt{\frac{(u_p - u'_p)^2 + (v_p - v'_p)^2}{W_{bbox_i} H_{bbox_i}}} \tag{11}$$

where i refers to any image sample, W_{bbox_i} and H_{bbox_i} are the width and height of the ground truth bounding box of the needle on that image sample.

As for 3D point localization, we use absolute L2 norm error, the unit is in mm.

$$L2.\,3D.\,err_p = \sqrt{(x_p - x'_p)^2 + (y_p - y'_p)^2 + (z_p - z'_p)^2} \tag{12}$$

As for 6 DOF pose estimation, we use absolute L2 norm error for the positional error. the unit is in mm.

$$pos.\,err = \sqrt{(x - x')^2 + (y - y')^2 + (z - z')^2} \tag{13}$$

where x, y, z are from the ground truth translation vector, x', y', z' are from the estimated translation vector.

Rotation error is computed as the angle of the rotation vector from the rotation error matrix $R_{err} = R \cdot R'$, R is the ground truth rotation matrix, and R' is the estimated rotation matrix.

$$ori.\,err = arccos(\frac{tr(R_{err}) - 1}{2}) \tag{14}$$

Table 1. Hyperparameters

Hyperparameter Name	Value
Epoch	20
Batch size	8
Learning rate	0.01
Momentum	0.9
Weight decay	0.0005

5. Results

5.1. *2D Keypoint Detection*

In order to examine the impact of the training set size on the neural network model's keypoint detection performance, we conducted 6 experiments using the auto-generated dataset containing a total of 6892 images, 1034 (15%) images were set aside for testing, and the remaining images were used for training purposes. To explore the relationship between the training set size and performance, we randomly selected subsets of 70 (1%), 345 (5%), 690 (10%), 2070 (30%), 3446 (50%), and 5858 (85%) images to train the models. Training hyperparamters are in Table. 1, and the training is performed on an NVidia A100 GPU.

During each experiment, we train the model for 20 epochs and saved the weights after each epoch, the weights with the least training loss are selected for evaluation on the same testing dataset. The results are presented in Table. 2. From it, we can see that as the size of the training set rises from 70 images to 2070 images, the keypoint detection accuracy was significantly improved (65.0%), however, when further more data (183%) are fed in, the accuracy gain was only 2.6%.

Table 2. Impact of training set size on keypoint detection error.

Training Set Size	Training Loss	L2 error	rel. L2 error
70 (1%)	7.79	83.96	69.8%
345 (5%)	4.26	31.17	26.7%
690 (10%)	3.52	18.14	15.5%
2070 (30%)	2.66	5.48	4.8%
3446 (50%)	2.23	3.64	3.1%
5858 (85%)	1.86	2.54	2.2%

5.2. *Point-wise Localization Error*

As we defined 5 key points along the needle, from a human perspective, it is relatively easy to identify and pick up the head point (A) and the tail point (E), but the key points in the middle (B, C, D) are more challenging to localize, as there are no clear local features or distinct markers associated with them. However, Table. 3 of our experiment results indicates that there is no large difference in localization accuracy among the different types of keypoints. Interestingly, the standard deviation for keypoints A and E is higher than that for B, C, and D, implying that the precision in localizing the middle points is higher than that for the head and tail points. While we think that using a human-labeled dataset would exhibit a different pattern, the localization errors here for all 5 keypoint are acceptable and better than the results we reported in,[7] with the help of a larger dataset of more various needle poses.

Table. 3 also demonstrates a strong correlation between the accuracy and precision of keypoint localization in 2D images and those in 3D space. So, for the pose estimation method based on stereo triangulation, improving the performance of 2D keypoint detection methods would likely result in a reduction of errors in 3D keypoint localization and thus in 6 DOF pose estimation.

Table 3. Point-wise keypoint localization error

Keypoint	2D rel. Error	2D SD	3D Error (mm)	3D SD
A	2.7%	0.082	3.2	5.5
B	2.3%	0.022	2.9	4.2
C	2.2%	0.019	2.9	2.9
D	2.1%	0.021	3.0	4.8
E	2.3%	0.072	3.2	11.5

5.3. *6DOF Pose Estimation*

Table. 4 presents the final 6 DOF pose estimation errors of our proposed methods. The median position error of the best pipeline is 1.3 mm, and the median orientation error is 2.9 degrees. When utilizing stereo images as input, the PnP-based method demonstrated a slight performance superiority over the stereo-triangulation method. Remarkably, the PnP-based approach was able to estimate the 6 DOF pose from a single 2D image with a comparable performance level to that achieved when using stereo images from either of the methods. Inference speed is up to 15 Hz on a personal computer with an NVidia 3080 GPU.

Table 4. Position and orientation tracking errors of the 120° needle, pink suturing phantom background (Fig. 1-a).

	Error Type	Mono-PnP	Stereo-PnP	PCR
Median	Position (mm)	1.37	1.28	2.13
	Orientation (°)	2.85	2.91	13.68
Avg.	Position (mm)	3.01	2.60	2.98
	Orientation (°)	19.97	19.16	21.56
Std.	Position (mm)	4.32	3.18	2.85
	Orientation (°)	42.29	30.60	24.03
RMSE	Position (mm)	5.27	4.08	4.12
	Orientation (°)	46.77	36.11	32.29

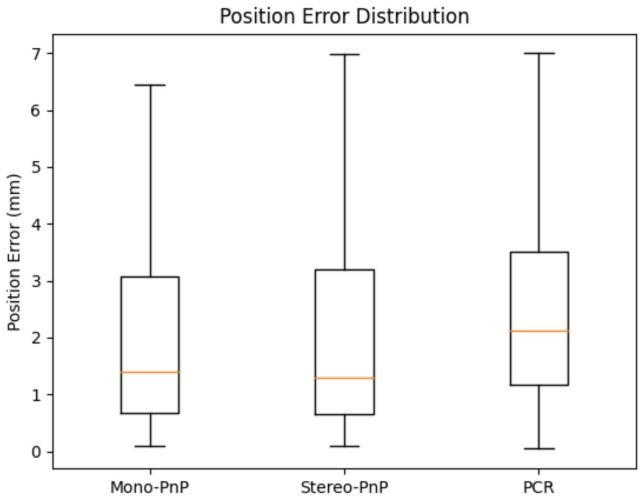

Fig. 7. Position Error Distribution. The boxes range from the first quartile to the third quartile.

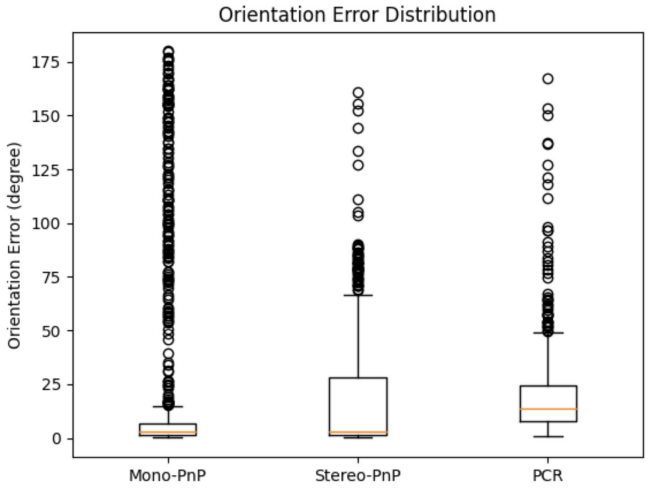

Fig. 8. Orientation Error Distribution. Fliers beyond 1.5x inter-quartile range from the third quartile are marked as circular dots.

5.4. *Multi-viewpoint Test*

From the pose estimation error distribution, we can see that the errors for a majority of the samples are small, but there are some outliers with large errors. Results from Sec. 4.2 as in Table. 5 and Table. 6 demonstrate that the average of multiple estimations from multi-viewpoint after eliminating the outliers can often reduce the errors. In all test cases, position errors were reduced with Multi-view, and there was only one case that the orientation error was increased, in Test 4. On average, the position error is reduced by 61% and the orientation error by 39%.

Table 5. Multi-viewpoint Experiment

Test	Error Type	Single View	Multi-view
Auto-sweep			
1	Position (mm)	2.4	1.3
	Orientation (°)	3.5	3.1
2	Position (mm)	1.4	0.5
	Orientation (°)	2.4	2.3
3	Position (mm)	3.2	0.6
	Orientation (°)	7.2	3.0
4	Position (mm)	1.6	0.8
	Orientation (°)	1.1	1.4
5	Position (mm)	0.4	0.3
	Orientation (°)	4.9	1.7
Manual			
1	Position (mm)	13.7	1.5
	Orientation (°)	146.5	1.2
2	Position (mm)	17.5	1.5
	Orientation (°)	13.6	0.9
3	Position (mm)	9.3	0.6
	Orientation (°)	98.4	1.3

Table 6. Comparison of position and orientation errors from single view and multi-view auto-sweep.

	Error Type	Single-view	Multi-view
Median	Position (mm)	1.6	0.6
	Orientation (°)	3.5	2.3
Avg.	Position (mm)	1.8	0.7
	Orientation (°)	3.8	2.3
Std.	Position (mm)	0.94	0.34
	Orientation (°)	2.10	0.67
RMSE	Position (mm)	2.03	0.77
	Orientation (°)	4.36	2.39

5.5. *Extensibility*

To further validate the extensibility of our proposed method, we conducted similar experiments with a new suture needle model of a different size and shape. Additionally, we varied the background by including the realistic 3-Dmed suturing pad and da Vinci PSMs. A dataset comprising 730 images was used for training, and the performance of the two pipelines was assessed on 70 test samples.

The results of the 6 DOF pose estimation are presented in Table 7.

Table 7. Position and orientation tracking errors of the 150° needle on varied backgrounds.

	Method	Mono-PnP	Stereo-PnP	PCR
Median	Position (mm)	1.54	1.45	3.12
	Orientation (°)	4.58	4.32	21.44
Avg.	Position (mm)	2.86	2.51	4.36
	Orientation (°)	13.43	12.32	28.73
Std.	Position (mm)	4.82	3.36	4.53
	Orientation (°)	23.52	17.60	23.33
RMSE	Position (mm)	5.61	4.19	6.29
	Orientation (°)	27.09	21.49	37.00

6. Discussion

Both of our proposed methods demonstrated good accuracy in estimating the position of the suture needle. However, when it comes to orientation estimation accuracy, the PnP (Perspective-n-Points) method outperformed the PCR (Point Cloud Registration) method. By including samples with varied backgrounds in training, our method maintained the same level of performance as it in a consistent background. Thus, to make the model perform well in a specific scenario, either including training samples under an environment similar to the testing environment, or incorporating domain randomization[35] is imperative.

The approach is not specific to any geometry constraints and can be extended to support various needle shapes used for suturing, even non-circular ones.) However, large inaccuracies were produced when the needle's body was obstructed, and in MIS, the needle is often grasped by tools, making it not fully visible to the camera. In this paper, our proposed method only aims to track a suture needle when the camera can see it completely. This is helpful at the beginning of a suturing task when a robot wants to grab it from a surface.

In terms of the multi-viewpoint idea, we wanted to find an early indicator to determine whether an estimation will be good or not, for identifying the outliers. The R-CNN model provides a confidence score that indicates the likelihood of an object being present inside the bounding box and the accuracy of the box itself. However, as long as the whole needle is present in the image, there is no correlation between the confidence score and the final pose estimation error. We also found that re-projection error is not strongly correlated with pose estimation errors.

7. Conclusion and Future Work

In summary, we designed a markerless tracking method that can estimate the 6 DOF pose of a suture needle precisely in the simulated surgical scene. A dataset of 6892 images for needle body points detection was programmatically generated and annotated, and we trained a Keypoint R-CNN model on it and achieved a low key points detection error of 2.2% relative to the size of the needle bounding box. We also investigated the impact of training dataset size on the keypoint detection performance. Two complete 6 DOF tracking pipelines were built using different point-to-point registration methods, and we achieved a median position error of 1.4 mm and a median orientation error is 2.9 degrees, We also utilized a robotic endoscope to ensemble transformations from multi-viewpoints, this idea was able to significantly reduced the pose estimation error from failed single view estimation cases. We also proved the extensibility of our approach by testing with a different needle model and varied background.

The simulation results show that our approach has the potential to be transferred to real-world cases. We are working on implementing this framework in a real-world scenario on the dVRK, but there are challenges. When preparing the training dataset, we rely on a very accurate needle pose and camera projection matrix to compute the true coordinates of the needle keypoints in an image. In reality, it is not feasible without attaching any marker or changing the appearance of the needle. So real-world dataset collection would be more challenging than that in simulation. One potential solution for this, similar to Thananjeyan et al. proposed in [36], is to mark the keypoints on the needle with ultraviolet-fluorescent paint, make those points only visible under ultraviolet light. Additionally, Transfer Learning techniques[37] will be utilized to make the most of the simulation data and mitigate the Sim-to-Real gap. To get the ground truth of the 6 DOF poses for evaluation purposes, we may still need to attach an external marker (optical/electromagnetic) to the needle and use a tracker, but the tracking pipeline will remain markerless. Furthermore, we also plan to include partially occluded needle images to the dataset, but modification on the R-CNN is necessary as in [38] proposed.

Acknowledgments

This work was supported in part by NSF AccelNet award OISE-1927275. Special thanks to Dr. Adnan Munawar and Prof. Peter Kazanzides for establishing the simulation framework and for organizing the 2021-2022 and 2023-2024 AccelNet Surgical Robotics Challenge. Thanks to Jack Bergin, Ellen Roberts, and Isaak Osterbur for their contribution to this project.

References

[1] A. Attanasio, B. Scaglioni, E. De Momi, P. Fiorini and P. Valdastri, Autonomy in surgical robotics, *Annual Review of Control, Robotics, and Autonomous Systems* **4** (2021) 651–679.

[2] S. Garrido-Jurado, R. Muñoz-Salinas, F. J. Madrid-Cuevas and M. J. Marín-Jiménez, Automatic generation and detection of highly reliable fiducial markers under occlusion, *Pattern Recognition* **47**(6) (2014) 2280–2292.

[3] O. Kedilioglu, T. M. Bocco, M. Landesberger, A. Rizzo and J. Franke, Arucoe: enhanced aruco marker, *2021 21st International Conference on Control, Automation and Systems (ICCAS)*, IEEE (2021), pp. 878–881.

[4] L. Qian, C. Song, Y. Jiang, Q. Luo, X. Ma, P. W. Chiu, Z. Li and P. Kazanzides, Flexivision: Teleporting the surgeon's eyes via robotic flexible endoscope and head-mounted display, *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE (2020), pp. 3281–3287.

[5] S. Gao, Y. Wang, X. Ma, H. Zhou, Y. Jiang, K. Yang, L. Lu, S. Wang, B. C. Nephew, L. Fichera *et al.*, Intraoperative laparoscopic photoacoustic image guidance system in the da vinci surgical system, *Biomedical Optics Express* **14**(9) (2023) 4914–4928.

[6] M. Allan, P.-L. Chang, S. Ourselin, D. J. Hawkes, A. Sridhar, J. Kelly and D. Stoyanov, Image based surgical instrument pose estimation with multi-class labelling and optical flow, *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part I 18*, Springer (2015), pp. 331–338.

[7] Y. Jiang, H. Zhou and G. S. Fischer, Markerless suture needle tracking from a robotic endoscope based on deep learning, *2023 International Symposium on Medical Robotics (ISMR)*, IEEE (2023), pp. 1–7.

[8] A. Munawar, J. Y. Wu, G. S. Fischer, R. H. Taylor and P. Kazanzides, Open simulation environment for learning and practice of robot-assisted surgical suturing, *IEEE Robotics and Automation Letters* **7**(2) (2022) 3843–3850.

[9] C. Wengert, L. Bossard, A. Häberling, C. Baur, G. Székely and P. C. Cattin, Endoscopic navigation for minimally invasive suturing, *International Conference on Medical Image Computing and Computer-Assisted Intervention*, Springer (2007), pp. 620–627.

[10] Y. Kurose, Y. M. Baek, Y. Kamei, S. Tanaka, K. Harada, S. Sora, A. Morita, N. Sugita and M. Mitsuishi, Preliminary study of needle tracking in a microsurgical robotic system for automated operations, *2013 13th international conference on control, automation and systems (ICCAS 2013)*, IEEE (2013), pp. 627–630.

[11] 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, *2016 IEEE international conference on robotics and automation (ICRA)*, IEEE (2016), pp. 4178–4185.

[12] S. Speidel, A. Kroehnert, S. Bodenstedt, H. Kenngott, B. Mueller-Stich and R. Dillmann, Image-based tracking of the suturing needle during laparoscopic interventions, *Medical Imaging 2015: Image-Guided Procedures, Robotic Interventions, and Modeling*, **9415**, SPIE (2015), pp. 70–75.

[13] C. D'Ettorre, G. Dwyer, X. Du, F. Chadebecq, F. Vasconcelos, E. De Momi and D. Stoyanov, Automated pick-up of suturing needles for robotic surgical assistance, *2018 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE (2018), pp. 1370–1377.

[14] M. Ferro, G. Fontanelli, F. Ficuciello, B. Siciliano and M. Vendittelli, Vision-based suturing needle tracking with extended kalman filter, *Computer/Robot Assisted Surgery workshop*, (2017).

[15] Q. Mei, J. Chainey, D. Asgar-Deen and D. Aalto, Detection of suture needle using deep learning, *Journal of Medical Robotics Research* **4**(03n04) (2019) p. 1942005.

[16] J. Redmon, S. Divvala, R. Girshick and A. Farhadi, You only look once: Unified, real-time object detection, *Proceedings of the IEEE conference on computer vision and pattern recognition*, (2016), pp. 779–788.

[17] R. Girshick, Fast r-cnn, *Proceedings of the IEEE international conference on computer vision*, (2015), pp. 1440–1448.

[18] M. Zhou, X. Wang, J. Weiss, A. Eslami, K. Huang, M. Maier, C. P. Lohmann, N. Navab, A. Knoll and M. A. Nasseri, Needle localization for robot-assisted subretinal injection based on deep learning, *2019 International Conference on Robotics and Automation (ICRA)*, IEEE (2019), pp. 8727–8732.

[19] A. Wilcox, J. Kerr, B. Thananjeyan, J. Ichnowski, M. Hwang, S. Paradis, D. Fer and K. Goldberg, Learning to localize, grasp, and hand over unmodified surgical needles, *2022 International Conference on Robotics and Automation (ICRA)*, IEEE (2022), pp. 9637–9643.

[20] S. Iyer, T. Looi and J. Drake, A single arm, single camera system for automated suturing, *2013 IEEE International Conference on Robotics and Automation*, IEEE (2013), pp. 239–244.

[21] Z.-Y. Chiu, A. Z. Liao, F. Richter, B. Johnson and M. C. Yip, Markerless suture needle 6d pose tracking with robust uncertainty estimation for autonomous minimally invasive robotic surgery, *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE (2022), pp. 5286–5292.

[22] O. Özgüner, R. Hao, R. C. Jackson, T. Shkurti, W. Newman and M. C. Cavusoglu, Three-dimensional surgical needle localization and tracking using stereo endoscopic image streams, *2018 IEEE international conference on robotics and automation (ICRA)*, IEEE

(2018), pp. 6617–6624.
[23] G. Wang, F. Manhardt, F. Tombari and X. Ji, Gdr-net: Geometry-guided direct regression network for monocular 6d object pose estimation, *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, (2021), pp. 16611–16621.
[24] K. He, G. Gkioxari, P. Dollár and R. Girshick, Mask r-cnn, *Proceedings of the IEEE international conference on computer vision*, (2017), pp. 2961–2969.
[25] Stanford Artificial Intelligence Laboratory et al., Robotic operating system.
[26] A. Munawar and G. S. Fischer, An asynchronous multi-body simulation framework for real-time dynamics, haptics and learning with application to surgical robots, *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE (2019), pp. 6268–6275.
[27] P. Kazanzides, Z. Chen, A. Deguet, G. S. Fischer, R. H. Taylor and S. P. DiMaio, An open-source research kit for the da vinci® surgical system, *2014 IEEE international conference on robotics and automation (ICRA)*, IEEE (2014), pp. 6434–6439.
[28] K. He, X. Zhang, S. Ren and J. Sun, Deep residual learning for image recognition, *Proceedings of the IEEE conference on computer vision and pattern recognition*, (2016), pp. 770–778.
[29] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, Pytorch: An imperative style, high-performance deep learning library, *Advances in neural information processing systems* **32** (2019).
[30] S. Ren, K. He, R. Girshick and J. Sun, Faster r-cnn: Towards real-time object detection with region proposal networks, *Advances in neural information processing systems* **28** (2015).
[31] M. A. Fischler and R. C. Bolles, Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography, *Communications of the ACM* **24**(6) (1981) 381–395.
[32] V. Lepetit, F. Moreno-Noguer and P. Fua, Epnp: An accurate o (n) solution to the pnp problem, *International journal of computer vision* **81**(2) (2009) 155–166.
[33] R. Shapiro, Direct linear transformation method for three-dimensional cinematography, *Research Quarterly. American Alliance for Health, Physical Education and Recreation* **49**(2) (1978) 197–205.
[34] K. S. Arun, T. S. Huang and S. D. Blostein, Least-squares fitting of two 3-d point sets, *IEEE Transactions on pattern analysis and machine intelligence* (5) (1987) 698–700.
[35] J. Tobin, R. Fong, A. Ray, J. Schneider, W. Zaremba and P. Abbeel, Domain randomization for transferring deep neural networks from simulation to the real world, *2017 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, IEEE (2017), pp. 23–30.
[36] B. Thananjeyan, J. Kerr, H. Huang, J. E. Gonzalez and K. Goldberg, All you need is luv: Unsupervised collection of labeled images using invisible uv fluorescent indicators, *arXiv preprint arXiv:2203.04566* (2022).
[37] F. Zhuang, Z. Qi, K. Duan, D. Xi, Y. Zhu, H. Zhu, H. Xiong and Q. He, A comprehensive survey on transfer learning, *Proceedings of the IEEE* **109**(1) (2020) 43–76.
[38] S. Zhang, L. Wen, X. Bian, Z. Lei and S. Z. Li, Occlusion-aware r-cnn: detecting pedestrians in a crowd, *Proceedings of the European Conference on Computer Vision (ECCV)*, (2018), pp. 637–653.

Yiwei Jiang received his M.S. degree in Robotics from Johns Hopkins University. Currently, he is a Ph.D. student and working as a Research Assistant in the Department of Robotics Engineering at Worcester Polytechnic Institute.
Yiwei is the author of over 5 technical publications. His research interests include Medical Robotics, Image-guided Therapy, Computer Vision and Machine Learning. He is an active member of the IEEE and IEEE Robotics and Automation Society.

Haoying Zhou received his M.S. degree in Mechanical Engineering from Boston University, USA. He is currently a Ph.D. student and working as a Research Assistant in the Department of Robotics Engineering at Worcester Polytechnic Institute, USA.

Haoying is the author of over 5 technical publications. His research interests include Medical Robotics, Surgical Robot Simulation and Teleoperation, Minimally Invasive Surgeries, Machine Learning, Deep Learning, Learning from Demonstrations. He is an active member of the IEEE and IEEE Robotics and Automation Societies.

Gregory S. Fischer received both his M.S. degree in Electrical Engineering and his Ph.D. degree in Mechanical Engineering from Johns Hopkins University, USA, in 2003 and 2008, respectively.

He is now the William Smith Dean's Professor and a faculty member in Robotics Engineering with appointments

in Mechanical Engineering and Biomedical Engineering at WPI, the founding director of the Automation and Interventional Medicine (AIM) Robotics Research Laboratory, and the Director of the MassTech-supported PracticePoint R&D Center for translational research in healthcare cyber-physical systems. Professor Fischer is also the founder and CEO of AiM Medical Robotics.

Professor Fischer's research interests include Medical robotics and computer-integrated surgery, and is the author of over 200 technical publications.