2023 International Symposium on Medical Robotics (ISMR)
Atlanta, GA, USA, April 19-21, 2023

Markerless Suture Needle Tracking From A Robotic Endoscope Based On Deep Learning

Yiwei Jiang[1], Haoying Zhou[1], and Gregory S. Fischer[1]

Abstract— **Advancements in robot-assisted surgery have been rapidly growing since two decades ago. More recently, the automation of robotic surgical tasks has become the focus of research. In this area, the detection and tracking of a surgical tool are crucial for an autonomous system to plan and perform a procedure. For example, knowing the position and posture of a needle is a prerequisite for an automatic suturing system to grasp it and perform suturing tasks. In this paper, we proposed a novel method, based on Deep Learning and Point-to-point Registration, to track the 6 degrees of freedom (DOF) pose of a metal suture needle from a robotic endoscope (an Endoscopic Camera Manipulator from the da Vinci Robotic Surgical Systems), without the help of any marker. The proposed approach was implemented and evaluated in a standard simulated surgical environment provided by the 2021-2022 AccelNet Surgical Robotics Challenge, thus demonstrates the potential to be translated into a real-world scenario. A customized dataset containing 836 images collected from the simulated scene with ground truth of poses and key points information was constructed to train the neural network model. The best pipeline achieved an average position error of 1.76 mm while the average orientation error is 8.55 degrees, and it can run up to 10 Hz on a PC.**

I. INTRODUCTION

Suturing requires high dexterity and is often a tedious task, especially in Minimally Invasive Surgeries (MIS), when surgeons manipulate tools to perform the procedure instead of using their hands. The suturing quality and completion time directly affect the success of the surgery and the patient's wellness. Recently, growing research interest is attracted to automating suturing with assistance from robotic systems, and advanced suturing is a technique that can be expected to reach Level 3 - Conditional Autonomy [1]. To realize automatic suturing with a robot, a fundamental prerequisite is that the robot needs to know where the needle locates and keeps tracking the pose precisely, so that the robot can pick up a needle and use it for suturing. Due to the potential of varying lighting conditions and complex tissue background during operation, tracking a small needle promptly irrespective of the surrounding environment is still a challenging problem.

In robot-assisted surgeries, visual feedback is usually available from a camera and plays a vital role in perceiving the scene. So, most of the needle tracking systems are developed based on the information from images, in the early years, researchers tried to use color-based and geometry-based methods to segment the needle body [2] and track the needle [3], [4], [5], [6], but their systems either require painting or attaching markers on the needle body, and need environment-specific tuning of parameters to accommodate the complex and varying surgical background with different objects and light conditions, textures, and lighting condition. More recently, advanced computer vision techniques including Deep Learning have been explored, Mei et al. ([7]) utilized two popular object detection architectures: You Only Look Once (YOLO) and R-CNN (Region-based Convolutional Neural Network) to extract the bounding box of a suture needle in the images, Zhou et al. [8] also used Feature Pyramid Net (FPN) to detect a tiny needle tip. However, their work all stays in 2D object detection/segmentation and does not include 6 DOF pose information. Wilcox et al. [9] combined semantic segmentation with random sample consensus (RANSAC) to obtain an estimated needle pose, but did not include a numerical evaluation of the accuracy.

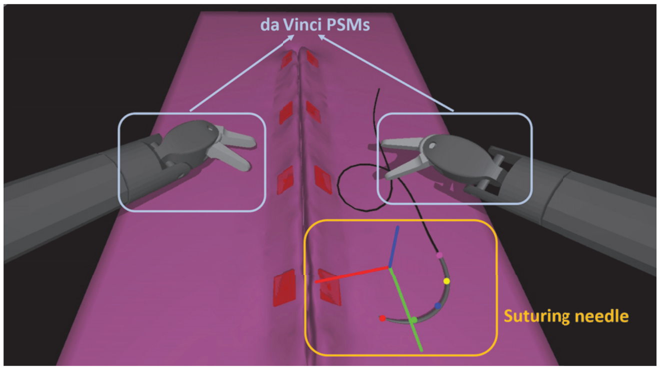

Fig. 1. Endoscope view of the simulated suturing scene provided by the 2021-2022 AccelNet Surgical Robotics Challenge, with the visualization of the needle pose tracking result from our implementation. PSM refers to Patient Side Manipulator.

Given the limitations of the existing literature mentioned above, in this paper, to the best of our knowledge, we are the first to propose a markerless method based on Deep Learning and Point-to-point Registration that only uses the information of the robotic endoscope to estimate the 6 DOF pose of a suture needle under a simulated surgical scene (Fig. 1). Our method does not require any modification to a commercial suture needle and introduces little interference to the existing surgical workflows. We constructed a customized dataset to train the neural network, tested the whole pipeline in a public, standard simulation environment by [10], and evaluated the accuracy and time efficiency.

This work was supported in part by NSF AccelNet award OISE-1927275.

[1] Yiwei Jiang, Haoying Zhou, and Gregory S. Fischer are with Department of Robotics Engineering, Worcester Polytechnic Institute, 100 Institute Rd, Worcester, MA 01609, USA. (emails: yjiang5, hzhou6, gfischer@wpi.edu)

2023 International Symposium on Medical Robotics (ISMR) | 979-8-3503-0162-5/23/$31.00 ©2023 IEEE | DOI: 10.1109/ISMR57123.2023.10130199

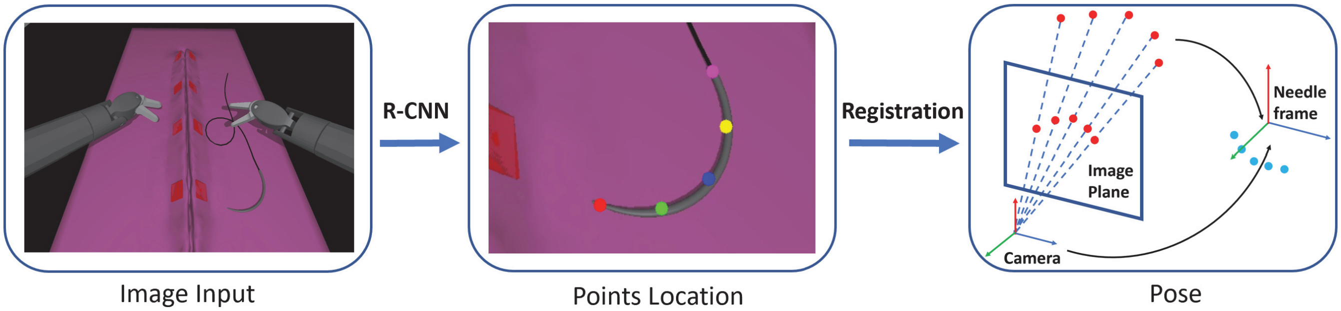

Fig. 2. Tracking method overview.

II. Methods

The goal of this work is to estimate the position and orientation of a suture needle solely from a robotic endoscope without the use of any markers. Briefly, our method has two steps, the first step is to extract the needle body points on a 2D image using a deep neural network architecture varied from Mask R-CNN [11], and the second step is to calculate the transformation using the positional information and the correspondence of the needle body points in different frames (Fig. 2). We also took advantage of the robotic endoscope to get multiple viewpoints of the needle and to reduce errors.

In the following sections, Section II. A introduces the environment in which we collected our training dataset and implemented our tracking system. Section II. B gives the details about the deep neural network that we used, and the training process. Section II. C illustrates the point-to-point registration methods we used to compute the transformation. Lastly, Section II. D illustrates how the robotic endoscope was utilized to include multi-viewpoint of the needle and to reduce the errors.

A. Data collection

The simulated surgical scene is provided by AccelNet Surgical Robotics Challenge [10], and it is built on Robot Operating System (ROS) [12] and Asynchronous Multi-Body Framework (AMBF) [13], it contains a suturing training phantom (in pink), the red squares on it are entry and exit holes for passing the needle, a needle with a thread connected to the tail, and a da Vinci surgical system patient side manipulator (PSM) from da Vinci Research Kit (dVRK) [14] with one Endoscopic Camera Manipulator (ECM), see Fig. 1.

The needle model used for the initial implementation (Fig. 3) is essentially a 120-degree arc with a radius of 10.18 mm, we defined five body points (A, B, ..., E) equally spaced on the arc. Note the coordinates of one of these points in the needle frame as $P_N = [x_N, y_N, z_N]$.

Ground truth poses of the needle and the ECM can be queried from ROS. The left and right cameras are rigidly fixed on the end-effector of the ECM. Note these transformations in the world coordinate system as T_{WN} (needle), T_{WE} (ECM end-effector), T_{WL} (left camera), and T_{WR} (right camera).

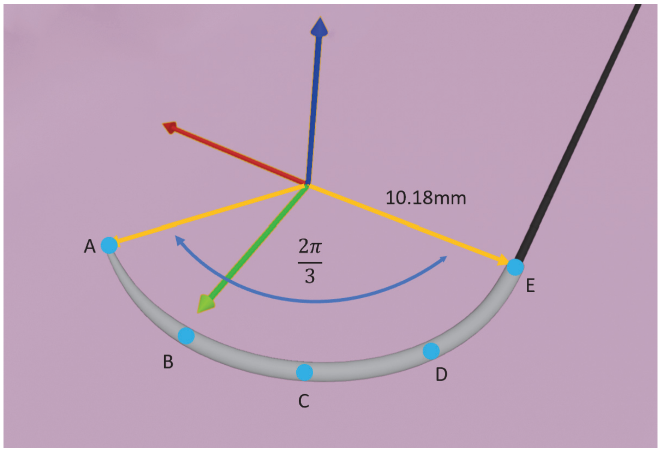

Fig. 3. Needle frame and keypoints on the needle body. All five of them are equally spaced on the arc, Point A is the tip of the needle, and Point E is the tail connected with a thread.

In this simulation, the virtual endoscope on the ECM acquires a 1080P stereo video stream at 30 FPS. On the collected images, the ground truth of the 5 key points locations was computed using the camera projection equations, and was used to train our neural network.

We collected 418 groups of data in total for training, each group of our dataset contains:

- a pair of images from the left and the right camera,
- 2D coordinates of the 5 needle body points in the left and right images,
- bounding box of the needle,
- transformation of the needle and the ECM in the world coordinate system.

Examples of our dataset are as Fig. 4 shows, including different ECM viewpoints and Patient Side Manipulator (PSM) poses to create different backgrounds but not to occlude the needle. We also varied brightness (±30%) and contrast (±30%), and rotated the images for data augmentation

B. R-CNN for Point Picking

The problem that extracting the 5 key points on the needle body lies in the Object Detection area in Deep Learning. Over the past few years, many Deep Learning models have been invented, such as YOLO [15], and R-CNN [16]. Since

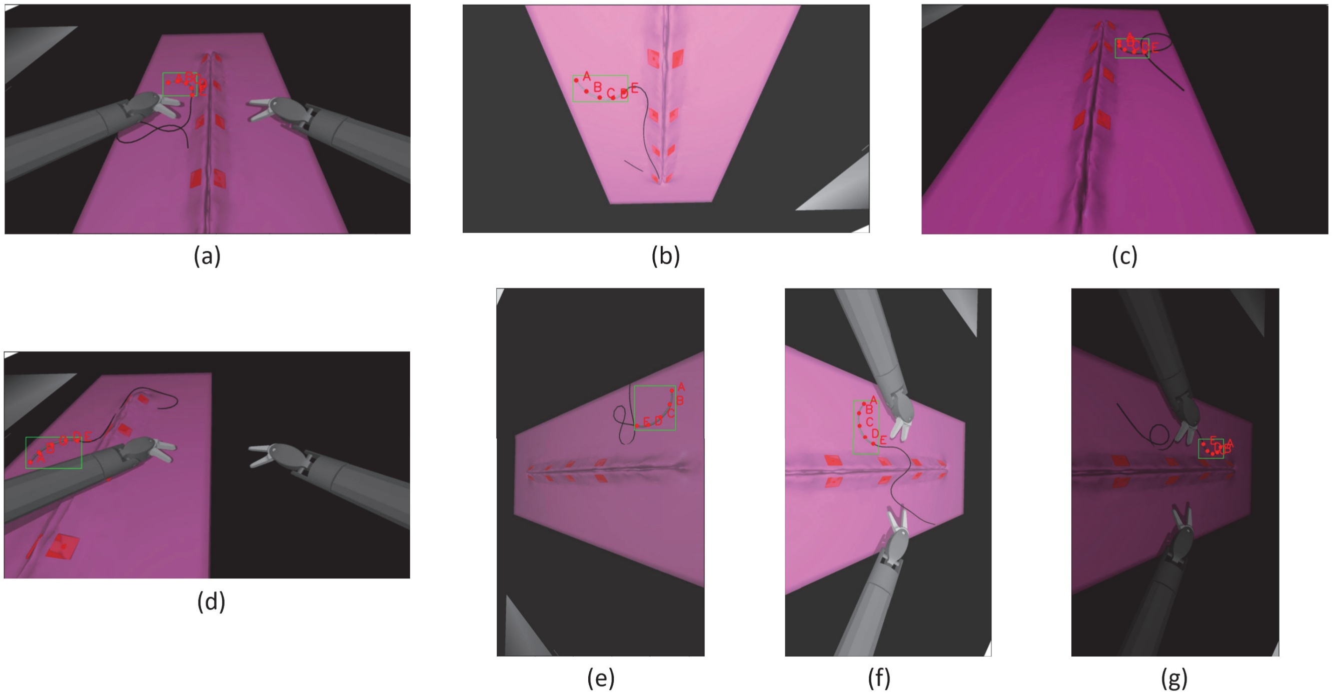

Fig. 4. Examples of the collected data. (a) A typical scene with default light settings, with both PSMs in the scene. (b) Flipped view, no PSMs in the scene, brighter, lower contrast. (c) Varied viewpoint, no PSMs, higher contrast. (d) Varied viewpoint, with PSMs. (e) Rotated view, no PSMs, dimmer, lower contrast. (f) Rotated view, with PSMs, lower contrast. (g) Rotated view, dimmer.

our aim is not only to obtain the bounding box of the needle but also the "landmarks" on the needle body, we choose to use a variant of Mask R-CNN [11] to achieve the goal. We implemented a Keypoint R-CNN model with a pre-trained ResNet-50-FPN [17] backbone using the PyTorch library [18]. Fine-tuned the model on a dataset with 836 images and tested it with 60 images. The loss function contains three terms, which are the classification and regression losses for both the Region Proposal Network and the R-CNN, and the keypoint loss (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), same as the loss function in [19] and [11].

$$Loss = L_{cls} + L_{reg} + L_{keypts} \tag{1}$$

C. Point-to-point Registration

Point-set registration is a process to find out a spatial transformation between two point sets in different coordinate systems. To find out the pose of the needle, we can use such a technique based on information about the needle body points from the previous step. Two pipelines were implemented, the first one is based on coordinates on the 2D images, and the second one makes use of stereo vision to get a 3D point set and then registers to the needle frame.

Pipeline I - 2D-to-3D Registration: Once we have the image coordinates of the needle body points from the previous step, and we also know the 3D coordinate of the points in the needle frame, estimating the pose of the needle with respect to the camera is so-called a Perspective-n-Point [20] problem. We used the Efficient Perspective-n-Point (EPnP) method [21] to directly calculate the 6 DOF transformation. The complete pipeline for one pair of images is as Fig. 5 shows. The final output is an average of the two transformations.

Pipeline II - 3D-to-3D Registration: Another way to calculate the pose is by 3D point set registration. Since we have a pair of images from the left and right camera, with the paired 2D coordinates of the body points and the camera parameters, we can use stereo triangulation to compute the 3D coordinates of these body points. Finally, given the shape of the needle and the local 3D coordinates of these body points, we can perform a point cloud registration (with known correspondence) to obtain the transformation from the camera frame to the needle frame.

For the Stereo Triangulation, we used Direct Linear Transformation (DLT) method [22]. Arun's method [23] was used to calculate the final transformation matrix from the point cloud in the camera frame to the point cloud in the needle frame. The complete pipeline that uses this registration method is as Fig. 6 shows.

D. Multi-viewpoint from a Robotic Endoscope

The ECM is a robotic arm with an endoscope, with the current joint values, the end-effector (endoscope) pose can be calculated from its 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 during a procedure. In suture

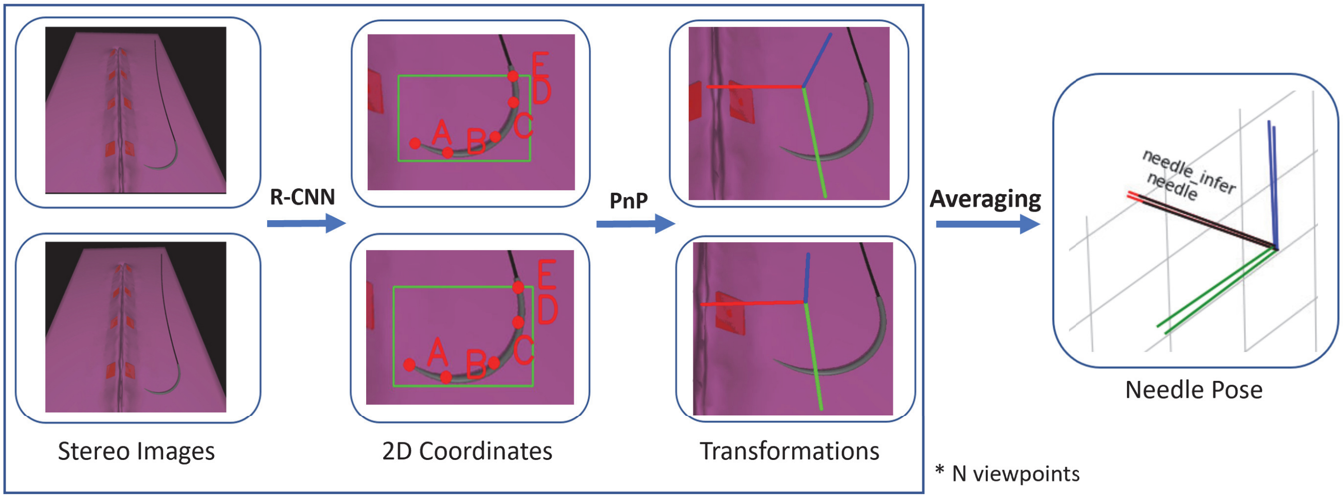

Fig. 5. Pipeline I. R-CNN extracts the defined 5 needle body points from the left and right images, the transformation from the left and right cameras to the needle are calculated respectively, the final result is an average of the two transformations. Transformations from multiple viewpoints can be combined, see Section II. D.

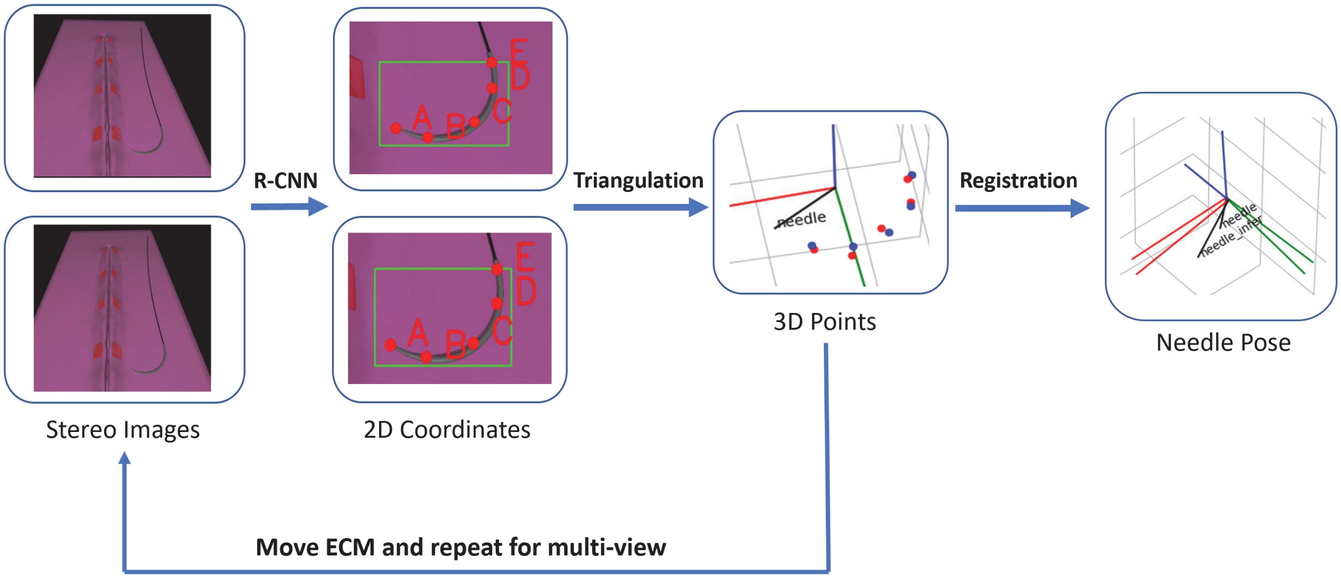

Fig. 6. Pipeline II. Both of the images are taken from the endoscope at the same time. 3D coordinates are triangulated from the 2D coordinates pairs (blue dots are the ground truth, red ones are from stereo triangulation). The needle pose is the 3D point sets registration result.

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 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 different multiple views of the needle. A valid image will be used for tracking only if the neural network reports a confidence score larger than a threshold (e.g., 0.5), which means that the endoscope probably sees the whole needle, then the transformation is computed via the EPnP approach. Multi-view tracking can also help to reduce random errors.

The workflow is as Fig. 7 shows. Transformations 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 change them to the current endoscope pose. The tracking result is the average of all elements in the queue after eliminating outliers.

III. Experiments and Results

The training hyperparameters are listed in Table I. To test the performance of our implemented methods, we randomly placed the needle on the suturing phantom with random poses

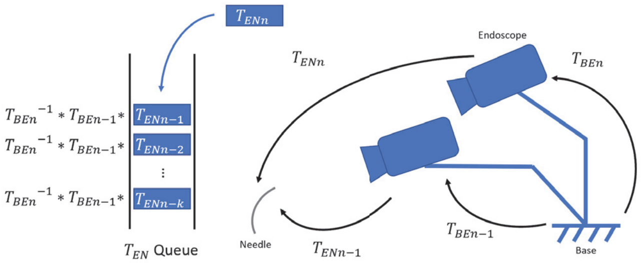

Fig. 7. ECM Multi-viewpoint ensemble. When the ECM moves from the previous pose (Index n-1) to the current pose (Index n), a new T_{EN} is estimated and inserted into the queue, all previous results ($T_{EN-1}, \ldots, T_{EN-k}$, k is the queue size) in the queue are multiplied by the ECM pose difference and are converted into the current endoscope frame.

TABLE I
HYPERPARAMETERS

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

30 times (60 images aside from the training dataset), the needle locations ranging from 10 cm to 25 cm away from the camera. Metrics including the 2D and 3D point localization errors and the 6 DOF pose estimation errors were evaluated.

A. Point Tracking

2D point-picking errors were computed as the Euclidean distance between our estimation and the ground truth. The tracking errors for 60 images of the 5 defined needle body points are shown in Table 2. We noticed that the error in the middle points (Point B, C) are relatively higher than the head point (A) and the tail point (E). We think it is because the middle points are harder to identify since there is no texture or marker on the needle surface. The overall average error is 5.58 pixels, and the image size is 1920x1080.

Pipeline II has one additional step for 3D point localization, the errors for 30 pairs of images of the 5 body points are also computed as the Euclidean distance between our results and the ground truth. The results are as Table 2 shows. 2D errors and 3D errors are strongly positively associated, with a Correlation Coefficient R = 0.63, which indicates that the 2D point picking accuracy largely affects the 3D localization.

TABLE II
POINT TRACKING ERRORS.

Keypoint	2D Error (pixel)	STD	3D error (mm)	STD
A	5.70	3.74	4.78	4.36
B	6.12	5.67	4.22	4.36
C	8.23	7.14	3.98	3.11
D	4.40	3.57	2.91	2.81
E	3.43	3.80	2.87	2.32

TABLE III
POSE TRACKING ERRORS.

Pipeline	I	II	I with Multi-view
Position (mm)	2.26	6.13	1.76
Orientation (degree)	6.74	35.61	8.55

B. Pose Estimation

The final pose errors of Pipeline I and II are shown in Table 3. Position errors were computed by the Euclidean distance between our estimation and the ground truth. For orientation, the errors are the magnitude of the Rodriguez vector representing the orientation from our estimation and the ground truth. Pipeline I is more accurate than Pipeline II. With the multi-viewpoint (one random joint of the ECM was moved for 0.05 rad each time) from the ECM, the position error was reduced to 1.76 mm. Fig. 8 shows the relationship between the tracking accuracy versus the distance from the endoscope to the target. Pipeline II is prone to large errors when the target is relatively far away from the camera.

C. Computational Efficiency

Our tracking system was running on a computer with an Nvidia RTX 3080 GPU and an Intel Core i7 12700K 12-Core Processor. The most computationally intensive part ($>$ 90%) is the forward propagation through the R-CNN model. On average in a test of 10 images, it takes 0.09 seconds to complete the process for each. The processing time for other steps is negligible (less than 0.01s). Thus, our pipeline can run up to 10 Hz on a relatively high-performance personal computer.

IV. DISCUSSION

Pipeline I and II share the same keypoint detection part but Pipeline I outperforms Pipeline II in overall pose tracking accuracy. Thus, we think stereo triangulation is more sensitive to 2D errors. Additionally, incorporating multi-viewpoint reduced the positional error a little but did not reduce the orientation errors, we think this may be because the keypoint detection part has systematic errors. Also, in such a simulated environment, random measurement error may not be a dominant factor.

The Keypoint R-CNN model was trained on a particular needle geometry. While the framework is not specific to any needle arc length or radius and should be robust to the variety of needle shapes used for suturing, more images with other shapes of needles need to be collected. Besides, 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 summary, we designed a markerless tracking method that can estimate the 6 DOF pose of a suture needle precisely in the simulated surgical scene up to 10 Hz, our

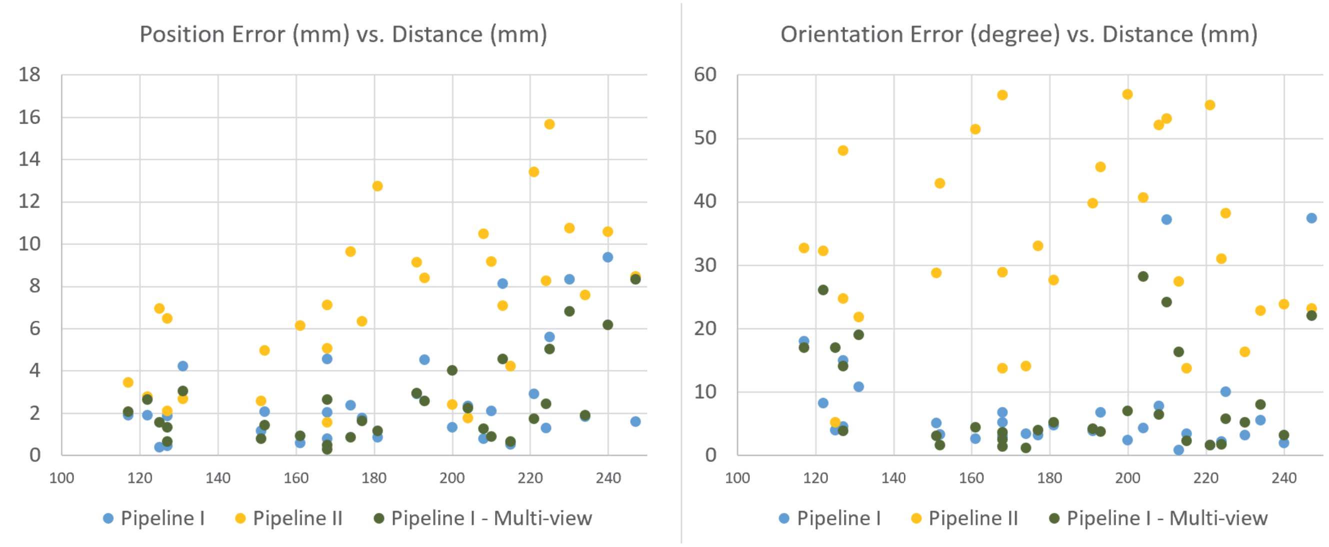

Fig. 8. Tracking error vs. Distance.

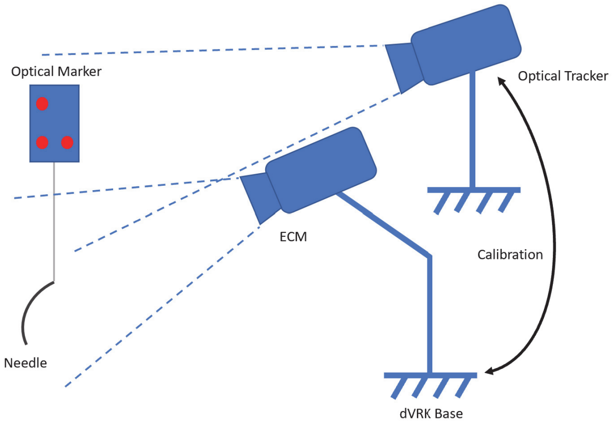

Fig. 9. A potential solution to obtain the ground truth of needle pose. An optical marker is rigidly mounted to a suture needle (through a rigid, thin, and transparent stick), the images taken by the robotic endoscope do not contain the marker. An optical tracker is calibrated with the dVRK base, so that the transformation between the ECM and the tracker can be computed by the calibration matrix and the forward kinematics of the ECM.

results indicate that this method has a good potential to be implemented in a real-world setup. A dataset for needle body points detection was produced and we trained a Keypoint R-CNN model on it and achieved a low relative key points detection error of 0.51%. Two complete tracking pipelines were built using different point-to-point registration methods, the average positional error is up to 2.26 mm and the average orientation error is up to 6.74 degrees. We also utilized a robotic endoscope to ensemble transformations from multi-viewpoints and reduced the average positional error to 1.76 mm.

V. Future Work

In terms of the limitations above, a variety of needle geometries will be included in our training dataset so the neural network model should be able to detect keypoints of needles of different shapes/sizes. Furthermore, we also plan to include partially occluded needle images to the dataset, but modification on the R-CNN is necessary as in [24] proposed.

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 [25], is to mark the keypoints on the needle with ultraviolet-fluorescent paint, make those points only visible under ultraviolet light. Additionally, Transfer Learning techniques [26] 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, as Fig. 9 shows, the tracking pipeline will remain markerless.

Acknowledgment

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 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*, vol. 4, pp. 651–679, 2021.

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

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

[4] 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 *2016 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2016, pp. 4178–4185.

[5] 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," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 1370–1377.

[6] 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," *arXiv preprint arXiv:2109.12722*, 2021.

[7] Q. Mei, J. Chainey, D. Asgar-Deen, and D. Aalto, "Detection of suture needle using deep learning," *Journal of Medical Robotics Research*, vol. 4, no. 03n04, p. 1942005, 2019.

[8] 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," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 8727–8732.

[9] 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," in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 9637–9643.

[10] 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*, vol. 7, no. 2, pp. 3843–3850, 2022.

[11] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2961–2969.

[12] Stanford Artificial Intelligence Laboratory et al., "Robotic operating system." [Online]. Available: https://www.ros.org

[13] A. Munawar and G. S. Fischer, "An asynchronous multi-body simulation framework for real-time dynamics, haptics and learning with application to surgical robots," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019, pp. 6268–6275.

[14] P. Kazanzides, Z. Chen, A. Deguet, G. S. Fischer, R. H. Taylor, and S. P. DiMaio, "An open-source research kit for the da vinci® surgical system," in *2014 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2014, pp. 6434–6439.

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

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

[17] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[18] 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*, vol. 32, 2019.

[19] 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*, vol. 28, 2015.

[20] 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*, vol. 24, no. 6, pp. 381–395, 1981.

[21] V. Lepetit, F. Moreno-Noguer, and P. Fua, "Epnp: An accurate o (n) solution to the pnp problem," *International journal of computer vision*, vol. 81, no. 2, pp. 155–166, 2009.

[22] R. Shapiro, "Direct linear transformation method for three-dimensional cinematography," *Research Quarterly. American Alliance for Health, Physical Education and Recreation*, vol. 49, no. 2, pp. 197–205, 1978.

[23] 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*, no. 5, pp. 698–700, 1987.

[24] S. Zhang, L. Wen, X. Bian, Z. Lei, and S. Z. Li, "Occlusion-aware r-cnn: detecting pedestrians in a crowd," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 637–653.

[25] 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.

[26] 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*, vol. 109, no. 1, pp. 43–76, 2020.