

Three-Dimensional Surgical Needle Localization and Tracking Using Stereo Endoscopic Image Streams

Orhan Özgüner, Ran Hao, Russell C. Jackson, Tom Shkurti, Wyatt Newman, and M. Cenk Çavuşoğlu

Abstract—This paper presents algorithms for three-dimensional tracking of surgical needles using the stereo endoscopic camera images obtained from the da Vinci[®] Surgical Robotic System. The proposed method employs Bayesian state estimation, computer vision techniques, and robot kinematics. A virtual needle rendering procedure is implemented to create simulated images of the surgical needle under the da Vinci[®] robot endoscope, which makes it possible to measure the similarity between the rendered needle image and the real needle. A particle filter algorithm using the mentioned techniques is then used for tracking the surgical needle. The performance of the tracking is experimentally evaluated using an actual da Vinci[®] surgical robotic system and quantitatively validated in a ROS/Gazebo simulation thereof.

I. INTRODUCTION

Minimally Invasive Surgery (MIS) has demonstrated significant benefits including reduced patient cost, length of hospital stay, recovery time, and postoperative pain [1]. The main goal of MIS is to duplicate open surgical techniques without large incisions in order to minimize complications. Advanced robotic systems such as the da Vinci[®] robotic surgical system (Intuitive Surgical, Inc., Sunnyvale, CA) allow surgeons to perform MIS using teleoperated robotic manipulators instead of manual surgical tools. Surgeons are provided with a console equipped with two master controllers that can manipulate robotic arms. The vision system provides a high-definition three-dimensional (3D) image for surgeons to see the surgical procedure in real-time, while the teleoperation system monitors the surgeon's hand movements and mimics with the robotic grippers with high precision.

Using such a system, surgeons are able to perform intricate surgeries with high precision and dexterity. Robotic minimally invasive surgery (RMIS) has been widely used in a variety of procedures [2], [3], including, prostatectomy [3], cardiac surgery [4] and thyroidectomy [5].

However, due to the nature of master-slave teleoperation, RMIS usually results in longer operation times and the learning curve is steeper than other MIS techniques [6]. In addition, the more limited robotic workspace and narrow laparoscopic camera view can make low-level tasks more challenging and time consuming. To enhance surgeon performance and reduce the operation time, autonomous robotic

surgical assistants [7] have been proposed to perform low-level surgical manipulation tasks such as suturing [8], [9], [10], debridement [11], dissection, and retraction [12].

Suturing is ideally suited to this form of automation due to its repetitive nature. To reduce the tissue trauma and operation time, an automation framework can keep the surgeon as the decision maker while relying on the robotic system to manage the execution of low-level motions.

The dynamic nature of surgical environments and the underlying substantial uncertainty necessitates the surgical manipulations to be performed under sensory guidance (as opposed to the use of primarily pre-planned manipulation strategies employed in traditional industrial robotics applications). As such, development of methods for perceiving the state of the surgical environment and the robotic system is a key requirement for autonomous and semi-autonomous execution of surgical manipulation tasks. Once such robust robotic perception algorithms are available, they can be used to perform precise visually guided manipulations by applying visual servo-control strategies, allow the robotic surgical system to operate under imperfect and varying robotic manipulator/camera calibration conditions, and to deal with the uncertainties resulting from unknown initial conditions and complex tissue deformation dynamics.

This study specifically focuses on the three-dimensional tracking of surgical needles from stereo endoscopic video image streams in RMIS. Localization and tracking of surgical needles is a key enabling technology for autonomous robotic execution of surgical suturing, since errors in needle trajectories resulting from incorrect needle grasps, needle slips, and robot/camera miscalibration may lead to task failures. For example, imperfect needle grasps may lead to increases in task completion time due to need for subsequent (possibly multiple) needle regrasps [13], whereas imperfect needle trajectories during a needle drive may cause increased tissue damage [14] or even result in complete failure of the suture. Knowledge of the pose of the surgical needle throughout the task would allow the system to deal with uncertainties in initial needle grasp as well as those resulting from unmodeled changes in the needle grasp configuration due to needle-tissue interaction forces.

We propose a vision-based probabilistic (Bayesian) state-estimation method for localizing and tracking the surgical needle. For state evolution, the proposed method employs the forward kinematics of the robotic surgical gripper manipulating the surgical needle whenever the needle is being directly grasped, and uses a Brownian motion model whenever the needle is not being directly grasped. Video images from

This work was supported in part by the National Science Foundation under grants CISE IIS-1524363 and CISE IIS-1563805, and National Institutes of Health under grant R01 EB018108.

The authors are with the Department of Electrical Engineering and Computer Science, Case Western Reserve University, Cleveland, OH. They can be reached via email at {oxo31,rxh349,rcj33,tes77,wsn,mcc14}@case.edu respectively.

stereo endoscopes provide the sensory feedback used in the measurement updates of the state estimation. A particle filter is used as the Bayesian estimator, as the underlying system is neither linear nor Gaussian. The proposed needle tracking algorithm is quantitatively validated by evaluating the needle tracking accuracy under different noise and occlusion conditions, using a ROS/Gazebo-based simulation of the da Vinci[®] surgical robotic system. The performance of the proposed algorithm is also validated in hardware experiments on an actual da Vinci[®] surgical robotic system for tracking a surgical needle and grasping it autonomously.

The rest of the paper is organized as follows. Section II discusses the related studies on surgical needle tracking. In Section III, the problem formulation and the proposed methods are introduced. The specific details of simulation and hardware based validation tests and the results of the needle tracking algorithm are presented in Section IV. The conclusions are presented in Section V.

II. RELATED STUDIES

The success of the autonomous suturing task is highly dependent on tracking task-critical elements such as the surgical needle, suture thread, and tissue. In order to complete the suturing task autonomously, the surgical needle needs to be localized and tracked during the execution of the task. Therefore, researchers studying the autonomous suturing task tried to solve this problem using various methods.

Nageotte et al. [15] proposed a circular needle and needle-holder localization algorithm for computer-assisted suturing in laparoscopic surgery. They modeled the needle-holder as a cylinder and attached passive visual markers to track its location. The surgical needle was colored to simplify its segmentation, and an ellipse-fitting algorithm was applied on the segmented image to calculate the 3D pose of the needle.

Sen et al. [16] developed an automated multi-throw surgical suturing algorithm using the da Vinci[®] surgical system. Instead of the da Vinci[®] endoscope, a custom stereo camera pair was employed to provide a larger workspace, and a larger-than-normal, yellow-painted surgical needle was used. In order to segment the image, the needle's distinctive shape and color were leveraged to assist HSV (Hue, Saturation, Value) segmentation to identify it. Once a set of image points are extracted from the segmented image, the 3D pose of the needle was calculated using an ellipse fitting algorithm.

Iyer et al. [17] attempted to solve the suturing problem by using a single-arm robotic manipulator with a standard laparoscopic needle holder, curved surgical needle and a clinical endoscope (single camera). In order to track the surgical needle, the obtained image was segmented using OpenCV thinning and smoothing functions. A monocular pose measurement method was then used along with the least-square ellipse fitting OpenCV functions to estimate the position and orientation of the semi-circular needle.

Kurose et al. [18] studied needle tracking for a microsurgery robotic system. They generated an off-line data set of 216,000 needle contour models using a 3D CAD model. During tracking, the image of the operation area was obtained

from the microscope and segmented using the Canny edge detection algorithm and needle color information. Then the segmented needle image was searched against the database of the needle contour models to find the best match.

The earlier studies in the literature on vision-based localization and tracking of surgical needles all rely on simplifying assumptions such as artificially colored needles, unchanged lighting conditions, a pregrasped needle, no tissue deformations, or custom built camera systems, none of which are applicable to practical RMIS scenarios. In contrast, the present study aims to perform needle tracking using images from the endoscopic stereo cameras of a realistic RMIS system without any modification to the robotic tools, endoscopes, or the surgical needles. Additionally, the goal is to achieve tracking which is robust to occlusions of the needle. Specifically, the availability of the state evolution model as part of the proposed Bayesian state-estimation scheme would allow the system to maintain tracking and to gracefully recover from periods of needle occlusion. To the best of authors' knowledge, there are no earlier published studies where detailed experiments were performed to test the repeatably and robustness of the tracking algorithm under occlusion, which is an inevitable scenario during RMIS.

III. PARTICLE FILTER ALGORITHM

Bayesian filtering algorithms enable estimation of a system's belief state under system and environmental uncertainties. In these algorithms, the belief ($bel(x_t)$) at time t is recursively calculated from the belief ($bel(x_{t-1})$) at time $t-1$ using measurement and control data as well as probabilistic models of the system dynamics and the measurement error. Bayesian state estimation methods have been used extensively in the robotics literature, and shown to be very suitable to solve real-time tracking problems [19], [20], [21].

We employ a particle filter as our underlying Bayesian filtering algorithm. The particle filter algorithm, also known as the sequential Monte Carlo estimation algorithm [22], [23], is one of the most popular Bayesian filtering techniques for nonlinear and non-Gaussian problems. The core idea of the particle filter algorithm is to approximate the posterior probability distribution of the state, such as the surgical needle pose, by using a finite number of randomly generated samples, called particles. Particle filters have been widely used in robotics and computer vision applications [22], including pose estimation on the $SE(3)$ group [24]. The particle filter algorithm can be divided into two main steps: state prediction and state update. In the prediction step, the particles are propagated using the system model. In the update step, the weight of each particle is measured based on the observation model. The overall flow diagram can be seen on Fig. 1, and the summary of the particle filter algorithm is given in Algorithm 1.

There are different approaches for generating the initial set of particles, such as randomly sampling the entire state space if there is no *a priori* information or generating samples in the region where the target is expected to be if there is *a priori* information. In this study, there are two cases: the

Algorithm 1: Particle filtering algorithm for surgical needle tracking. $\mathfrak{B}_t = \{x_t^{[n]}, n = 1 \dots N\}$ represents the belief of the needle state at time t . N is the number of particles used. u_t and z_t are respectively the control input to system and the observed measurement at time t .

```

1 function UpdateNeedleBelief( $\mathfrak{B}_{t-1}, u_t, z_t$ )
2    $\tilde{\mathfrak{B}}_t \leftarrow \emptyset$ 
3    $\mathfrak{B}_t \leftarrow \emptyset$ 
4   for  $n = 1 \rightarrow N$  do
5     sample  $x_t^{[n]} \sim p(x_t | u_t, x_{t-1}^{[n]})$ 
6      $\omega_t^{[n]} \leftarrow p(z_t | x_t^{[n]})$ 
7      $\tilde{\mathfrak{B}}_t \leftarrow \tilde{\mathfrak{B}}_t + (x_t^{[n]}, \omega_t^{[n]})$ 
8   end
9    $\mathfrak{B}_t \leftarrow \text{LowVarianceResampler}(\tilde{\mathfrak{B}}_t)$ 
10 return  $\mathfrak{B}_t$ 
11 end function

```

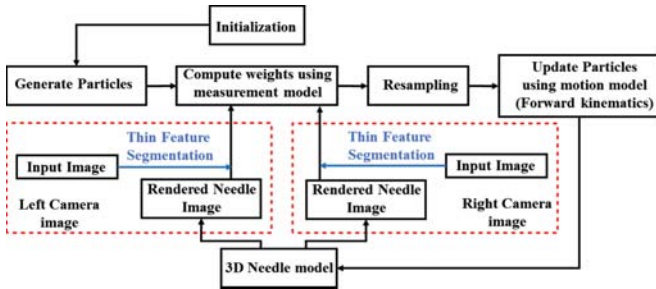


Fig. 1. Proposed particle filter algorithm flow for surgical needle tracking.

needle is either initially held by the robotic hand or it is freely placed in the surgical scene. If the needle is grasped by the robot, the forward kinematics of the robot arm are used to generate the initial set of samples around the expected needle pose. If the needle is free, then it is assumed that the rough pose of the needle is known, and the initial set particles are randomly generated in this region.¹

The first step in the algorithm (Algorithm 1, Line 5) is to update the particle states using the motion model. In this study, there are two different cases for the update step. If the needle is free, i.e., not being held by the robot, then the motion of the needle is modeled as Brownian motion, where the state of each particle is perturbed at each time step with Gaussian noise. This model would allow the particle filter to follow a (slowly) moving needle, as well as to accommodate uncertainties in the initialized needle pose. If the needle is grasped by the robot, then each particle state is updated using the incremental motion of the robotic

¹This latter assumption is used to avoid excessive computation necessary to solve the global localization problem at the initialization stage. This is not a severely restrictive assumption, since it is not unreasonable for a supervisory-controlled intelligent robotic surgical assistance system to expect the physician to indicate the general location of the needle to be used for each specific surgical task. Alternatively, it is also possible to construct a cold-initialization algorithm that employs computer vision techniques to search and locate the needle in the scene. However, this latter approach has not been pursued in this study.

gripper holding the needle. The incremental motion of the gripper can be estimated using the forward kinematics or the Jacobian of the manipulator, with added Gaussian noise to account for uncertainties in robot motion as well as errors in robot calibration.

The next step in the algorithm is the measurement update (Algorithm 1, Line 6). In the particle filter, each particle represents a hypothesis for the position and orientation of the needle. As part of the measurement update, the observation likelihoods for each of these hypotheses need to be evaluated. The observation likelihoods quantify how well each of the needle pose hypotheses match the observed images of the surgical scene captured by the endoscopic stereo vision system. In our proposed approach, the images acquired by the endoscope are segmented using a thin feature extraction algorithm in order to emphasize the needle outline (and other thin features in the scene). The observation likelihoods are then estimated from the image space similarity between the virtual images of the needle geometry generated from the needle pose hypotheses and the observed segmented images of the scene, as calculated using the normalized cross-correlation between them.

The final step of the algorithm is the resampling step (Algorithm 1, Line 9). In our proposed approach, the low variance resampling method [25] is employed.

A. Thin Feature Segmentation

Due to the thin and cylindrical structure of the surgical needle, many edge detection algorithms detect the surroundings of the needle instead of the needle itself. In order to segment the surgical needle more reliably, a thin feature enhancement algorithm developed by Jackson et al. [10], [26], which is based on algorithms from Frangi et al. [27] and Steger et al. [28], was employed.

The input image from the camera is first converted into grayscale then convolved with the Hessian matrix. This convolution produces a matrix for each pixel of the original gray scale image. Each matrix contains a set of 4 pixels that is in the form of a 2×2 matrix in $\mathbb{R}^{2 \times 2}$ for each pixel.

$$H(\sigma) = \begin{bmatrix} \frac{\partial^2}{\partial x^2} & \frac{\partial^2}{\partial x \partial y} \\ \frac{\partial^2}{\partial y \partial x} & \frac{\partial^2}{\partial y^2} \end{bmatrix} (\mathcal{N}_2(\sigma^2, x, y) * I) \quad (1)$$

where I is the original gray scale image and $\mathcal{N}_2(\sigma^2, x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{\sigma^2}}$ is a 2 dimensional Gaussian kernel. Each Hessian matrix is composed of eigenvalues ($|\lambda_1| < |\lambda_2|$) and orthogonal eigenvectors $v_1 \perp v_2 \in \mathbb{R}^2$. The eigenvalues correspond to the two principle directions as they relate to the Hessian of the convolved scale space image representation. The output image is generated using the magnitude and ratio of these eigenvalues. The local image region likely contains a thin feature, such as a surgical needle, when one of the eigenvalues is large and the other is small. Using the eigenvalues of the corresponding Hessian matrix, the maps $V'_l : \Omega \rightarrow \mathbb{R}^2$ for each pixel location $V'_l(i, j)$ are generated as follows.

$$\|V'_l\| = e^{-\frac{\sigma_l^2}{2\beta^2}} (1 - e^{-\frac{s^2}{2c^2}}) \quad (2)$$

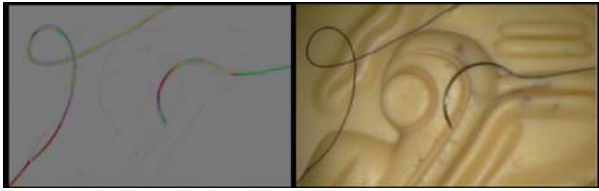


Fig. 2. Segmentation algorithm resulting image (left) using the original stereo camera image (right).

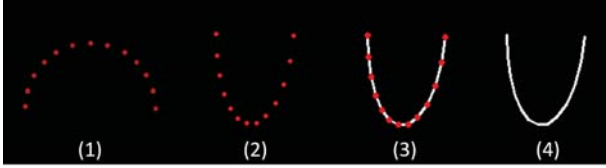


Fig. 3. Two dimensional points are sampled around the semicircular shape. Points are then projected onto the image space using the camera intrinsic matrix. Image space points are connected with lines to shape the needle geometry in the camera image space.

$$V'_l = v_2 \|V'_l\| \quad (3)$$

where $s = \sqrt{\lambda_1^2 + \lambda_2^2}$ and $\mathfrak{R}_B = |\lambda_1|/|\lambda_2|$ are measures of how small λ_2 is compared to λ_1 . The parameters β and c are user defined parameters to tune the segmentation result based on the camera and lighting conditions. In order to reduce the image processing noise, the map $V'_l(i, j)$ is smoothed using a normalized low pass Gaussian kernel. The final results of the segmented images for the left and right stereo images are obtained as $V_l, V_r : \Omega \rightarrow \mathbb{R}^2$ which is shown in Fig. 2. In order to increase the speed of the segmentation process, a graphical processing unit (GPU) is employed through the CUDA interface.

B. Virtual Needle Rendering

One of the main components of a Bayesian state estimation scheme is the measurement model. As part of the measurement model, a method to virtually render a surgical needle is employed to represent a needle pose hypothesis as observed through the stereo endoscope cameras of the system (described in Section II.C.). The surgical needle is modeled as a semicircle with a fixed radius.

The surgical needle is rendered using the camera projection matrix as shown in Fig. 3. First, 3D points are sampled that form a semicircle in the needle frame (1). Then points are projected onto the image space using the camera intrinsic matrix (2). The projected points form the two-dimensional needle shape in the image space. Finally points are connected with lines (3) to form the needle image in the image space (4). Increasing the number of sample points result in a smoother needle shape while reducing the overall execution speed of the algorithm since the number of projections also increases.

C. Measurement Model

In this study, the measurement model quantifies the similarity between the segmented image received from the

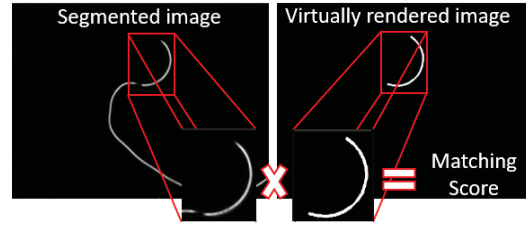


Fig. 4. The box containing the rendered image is saved during the rendering procedure. Using the box, segmented and rendered images are cropped from the same locations and used as the template and the source images in the template matching algorithm with a normalized cross correlation pixel-wise matching metric.

camera stream and the virtually-rendered hypotheses of the two-dimensional needle pose image. Normalized cross-correlation (available through the OpenCV template matching algorithm) is used to measure the similarity:

$$R = \frac{\sum_{x', y'} T(x', y') I(x', y')}{\sqrt{\sum_{x', y'} T(x', y')^2 I(x', y')^2}} \quad (4)$$

The weight $\omega_i^{[n]}$ of a given particle n is then set as

$$\omega_i^i := \sqrt{R_{left}^2 + R_{right}^2} \quad (5)$$

where R_{left} and R_{right} are the matching scores calculated using the left and right camera images, respectively.

D. Motion Model

When tracking a surgical needle, the motion of the real object needs to be modeled. In this study, the motion model is separated into two cases: one where the needle is free and one where the needle is held by the manipulator. If the needle is free, then there is no motion to be observed from the manipulator sensors. In this case, the motion of the needle is modeled as Brownian motion, perturbing the particles with a small amount of Gaussian noise at every iteration. This perturbation motion spreads the particles to avoid false convergence and enables a search mechanism. Once the free needle is localized, it can be grabbed by the manipulator at its now known configuration.

In the second case, the needle needs to be tracked while being held and moved by a manipulator. The coordinate transformations that will be used for the derivation of the motion model for this case are shown in Fig. 5.²

$$V_{CN}^b = Ad_{g_{SN}} V_{CS}^b + V_{SN}^b \quad (6)$$

where V_{CN}^b is the body velocity of the needle in terms of the camera frame, V_{SN}^b is the body velocity of the needle in terms of the manipulator frame and V_{CS}^b is the body velocity of the manipulator in terms of the camera frame. Since

²For simplicity, the camera is assumed to be at a fixed configuration relative to the robotic manipulator base frame in the following derivation. However, the formulation can easily be generalized to the moving camera case.

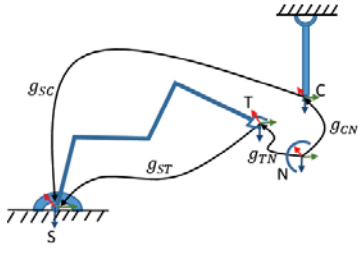


Fig. 5. Frames and transformation matrices used in the motion model derivation, where g_{SC} is the transformation between the camera and the manipulator base, $g_{SC}(\theta(t))$ is the tool tip transformation calculated from forward kinematics at time t , g_{CN} is the configuration of the needle frame in terms of the camera frame and g_{TN} is the needle grasp configuration, which is assumed to be known (see below).

the transformation between the camera and the manipulator is fixed, the body velocity of the camera in terms of the manipulator is $V_{CS}^b = 0$. Then, the body velocity of the needle, in terms of the camera frame, is the same as the body velocity of the needle, in terms of the manipulator frame:

$$V_{CN}^b = V_{SN}^b. \quad (7)$$

Using the robot kinematics, the body velocity of the needle, in terms of the manipulator frame, can be calculated as

$$V_{SN}^b = Ad_{g_{TN}^{-1}} V_{ST}^b + \underbrace{V_{TN}^b}_{\mathbf{0}}, \quad (8)$$

where V_{SN}^b and V_{ST}^b are respectively the body velocities of the needle and manipulator tip, in terms of the manipulator frame, and V_{TN}^b is the body velocity of the needle, in terms of the manipulator tip frame. Here, the needle is grasped by the manipulator with the configuration of g_{TN} . This configuration is assumed to be known and to remain constant during the motion. Therefore, the body velocity of the needle, in terms of the manipulator tip frame, is $V_{TN}^b = 0$.

The body velocity of the tool tip in terms of the manipulator frame is computed using the manipulator Jacobian as

$$V_{ST}^b = J_{ST}^b \dot{\theta}_{robot}, \quad (9)$$

where J_{ST}^b is the robot's body Jacobian and $\dot{\theta}_{robot}$ is a vector containing the sensor readings of the robot joints. Using (7), the body velocity of the needle in terms of the camera frame can be written as:

$$V_{CN}^b = Ad_{g_{TN}^{-1}} J_{ST}^b \dot{\theta}_{robot}. \quad (10)$$

In order to find the displacement of the real needle in space, the spatial velocity of the needle, in terms of the camera frame, V_{CN}^s , is computed from the body velocity of the needle in terms of the camera frame, V_{CN}^b :

$$V_{CN}^s = Ad_{g_{CN}} V_{CN}^b, \quad (11)$$

where g_{CN} is the transformation between the needle and the camera. Finally, each particle is updated as:

$$g_{CN}(t + \Delta t) = e^{\hat{V}_{CN}^s \Delta t} g_{CN}(t), \quad (12)$$

where $e^{\hat{V}_{CN}^s \Delta t}$ is the needle displacement predicted by the motion model, and

$$\hat{V}_{CN}^s = V_{CN}^s + \mathcal{N}(\Sigma) \quad (13)$$

is the spatial velocity of the needle, in terms of the camera frame, with additive Gaussian noise $\mathcal{N}(\Sigma)$ to account for robot motion uncertainties.

IV. VALIDATION RESULTS

The particle filter algorithm for the surgical needle localization and tracking was evaluated both in a simulation environment and on the real robot system. In the simulation and hardware tests, a 26 mm diameter semi-circular surgical needle is being localized and tracked by the proposed algorithm using images from the stereo endoscopic cameras of the da Vinci[®] surgical robotic system.

A. Simulation-based Validation Results

The proposed method was validated using a ROS/Gazebo-based simulation of the da Vinci[®] surgical robotic system. This simulation environment provides a perfect baseline since the camera to robot transformation, robot forward kinematics, and the joint sensor feedback are exactly known. In order to create a validation scenario, noise is added to the joint sensor feedback to the tracking algorithm so that the position and orientation values are more realistic.

In the first set of examples, the static convergence of the particle filter algorithm is validated. Specifically, the needle stays in the same pose, while the initial particle set is initialized with several different levels of uncertainty. In these examples, the particle filter algorithm is initialized with 250 particles in the Gazebo world.

Examples of surgical needle localization results can be seen in Fig. 6 and 7. In the top row of the example shown in Fig. 6, the best matched particle is projected onto the Gazebo endoscopic images in cyan. The bottom row images show the thin feature segmented camera images with the distribution of all of the particles. In this example, particle

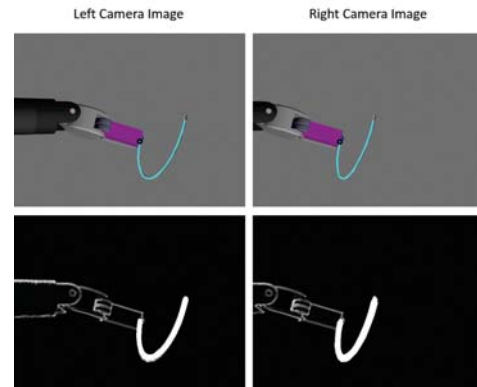


Fig. 6. An example surgical needle tracking result in Gazebo-based simulation environment. Top row images show the tracking result of the best particle rendered as cyan. Bottom row images show the segmented images superimposed with the distribution of all of the particles. In both cases, the left and right camera views are given in the corresponding images.

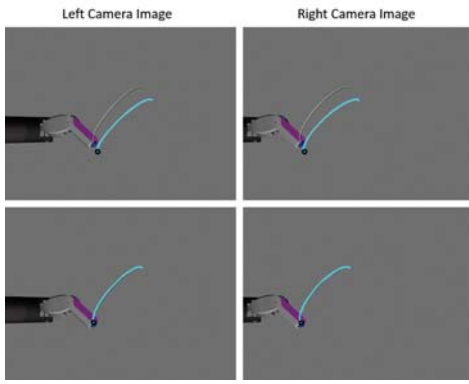


Fig. 7. Example surgical needle tracking result in Gazebo-based simulation environment: Top row images show the initial particle error and bottom row images show the best matched particle. In this example, the particle filter algorithm is using 250 particles. Initial position and orientation errors were 6mm and 5° respectively. When the particles converged, the resulting position and orientation errors were 0.62mm and 0.96° respectively. In both cases, the left and right camera views are given in the corresponding images.

filter was initialized with an initial position error of 7mm and orientation error of 8° . In this scenario, the particles converged with a final position error of 0.7mm and the orientation error of 2.3° . A second example of localization result is presented in Fig. 7, where the initial and final errors of the needle pose (as estimated by the best particle) are shown respectively in the top and bottom row of images.

The static localization performance of the proposed method was tested using 50 randomly selected needle positions and orientations under the restriction that the needle must be visible to the camera without occlusion, and each position was tested with four different levels of initial position and orientation errors. For each pose of the needle, the best particle position and orientation error were recorded at each step of the algorithm. The results of the average of 50 poses for the three smallest levels of initial error are shown in Fig. 8.

In Fig. 9, the localization errors for each of the 50 random trials for the highest initial position and orientation error case (8.7mm position and 9.16° orientation error) are presented. It can be seen that the particle filter algorithm did not converge at a few of the poses. This is not unexpected, since, in particle filter algorithms, as the initial uncertainty increases, more particles are needed to achieve sufficient coverage of the state space for proper convergence.³

In the second set of validation results, the dynamic tracking performance of the proposed algorithm are evaluated. In order to test the robustness of the proposed method during tracking, the surgical needle was grasped by the robot arm and followed 40 randomly selected needle trajectories in the Gazebo simulation environment. At each trajectory, the particles were initialized with the true position and orientation of the simulated needle using 250 particles. Three different noise levels were added to the joint sensor feedback during the motion of the arm. The position and angle errors

³There were no convergence problems encountered at the three lower uncertainty levels evaluated.

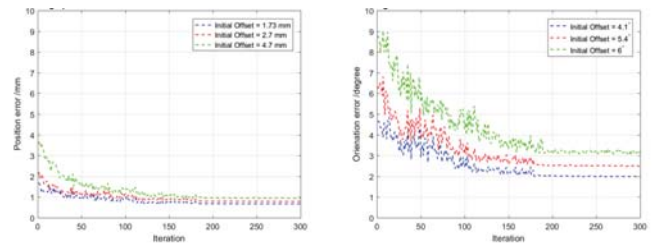


Fig. 8. The particle filter algorithm is initialized with 250 particles and tested using 50 different randomly selected poses with a maximum position and orientation error of 8.7mm and 9.16° respectively.

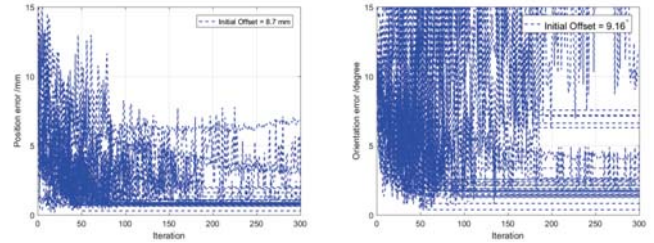


Fig. 9. Particle filter algorithm tested with initial position and orientation error of 8.7mm and 9.16° respectively. Localization errors for 50 different randomly chosen needle configurations are shown.

averaged over the 40 trials for each of the three noise level cases are presented in Fig. 10.

The simulation-based validation results demonstrate that the surgical needle tracking algorithm using a particle filter is robust for different levels of initial position errors as well as the joint sensor feedback errors. The initial location guess of the needle has a large impact the convergence time and in some cases the likelihood of success.

B. Hardware Validation Results

The proposed method was validated on a da Vinci[®] IS-1200 Surgical Robotic System, upgraded with the open-source/open-hardware da Vinci Research Kit (dVRK) [29], which allows direct computer-based control of the system.

The performance of the proposed method is tested under different conditions including occlusions. The surgical needle localization results using 300 particles under six different

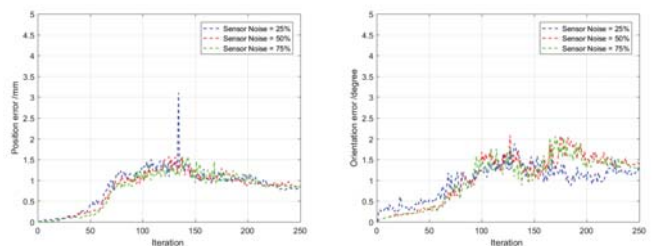


Fig. 10. Particle filter algorithm performance under three different levels of joint sensor noise. The algorithm was tested with 40 different initial poses, each of which followed a different randomly generated trajectory. The best particle position and orientation error were recorded at each iteration for 40 trajectories. The plots show the position and angle errors averaged over the 40 trials for each of the three noise level cases.

needle position and orientation cases are shown in Fig. 11. In these examples, the surgical needle is tracked while it is being held by one of the da Vinci[®] robot manipulators. In order to create a realistic scenario for RMIS, the second da Vinci[®] robot manipulator are manually inserted into the scene. To increase the background complexity, another needle is held by the second da Vinci[®] robot manipulator and presented in a similar position and orientation so that it could create a potential false positive match. This scenario was specifically created to test if the proposed method is able to handle potential local optima in the measurement model.

The experimental results indicate that the needle tracking algorithm can handle occlusions and potential false positive disturbances. It is also observed that the algorithm is able to recover the needle position even in a complex environment.

The tracking performance of the proposed method was further tested while the needle was in motion. For this experiment, the needle was held by the da Vinci[®] robotic manipulator with a known configuration and moved under the endoscope. During the tracking, the second tool holding another needle was presented into the scene. Fig. 12 shows a time sequence of the needle tracking algorithm results where the needle is occluded in different variations. The results indicate that the tracking algorithm can successfully track the surgical needle under occlusions and that it can gracefully recover from intermittent occlusions. A video attachment of the presented paper shows additional hardware tracking results.

Fig. 13 presents a physical validation of the algorithm while performing a needle grasp task. The particle filter algorithm is initialized with 300 particles sampled randomly by covering the workspace. A surgical needle is randomly placed into the scene. The algorithm then localizes the needle position and orientation. After the convergence of the localization, the needle position and orientation are sent to the robot controllers to successfully grasp the needle.

V. CONCLUSIONS

This study presents a particle filter-based framework for tracking surgical needles using stereo endoscopic images. A virtual needle rendering procedure is implemented to produce candidate needle models in the image space. Using the virtually rendered tool model, an image matching method is applied to form a measurement model for the particle filter algorithm. The performance of the tracking is quantitatively validated in a simulation environment and experimentally evaluated using the physical da Vinci[®] surgical robotic system. The validation results indicate that the proposed algorithm can successfully localize the needle starting from moderate levels of initialization uncertainty, and successfully track the needle while it is being manipulated by a robotic gripper. The algorithm was also able to maintain tracking in complex scenes, and under intermittent occlusions of the needle.

Future work will proceed in several directions. The proposed method uses the segmentation method implemented on a GPU-based parallel computing scheme. However, other

components of the algorithm are implemented as a serial algorithm on a CPU. Although the tracking algorithm runs in real-time, the frame rate (3 frames per second for 250 particles) is currently insufficient for closed-loop visual servo control. The most time consuming component of the tracking algorithm is the virtual tool rendering part since the needle frame points projected into the image space several times for each particle. As a part of the future work, we are working on a GPU-based parallel implementation of the particle filter algorithm to increase the speed of the algorithm up to a sufficient frame rate for a closed-loop visual servo control.

REFERENCES

- [1] P. Miccoli, R. Bellantone, M. Mourad, M. Walz, M. Raffaelli, and P. Berti, "Minimally invasive video-assisted thyroidectomy: Multi-institutional experience," *World Journal of Surgery*, vol. 26, no. 8, pp. 972–975, 2002.
- [2] A. R. Lanfranco, A. E. Castellanos, J. P. Desai, and W. C. Meyers, "Robotic surgery: A current perspective," *Annals of Surgery*, vol. 239, no. 1, pp. 14–21, 2004.
- [3] J. A. Smith Jr and S. D. Herrell, "Robotic-assisted laparoscopic prostatectomy: Do minimally invasive approaches offer significant advantages?" *Journal of Clinical Oncology*, vol. 23, no. 32, pp. 8170–8175, 2005.
- [4] F. W. Mohr, V. Falk, A. Diegeler, T. Walther, J. F. Gummert, J. Bucerius, S. Jacobs, and R. Autschbach, "Computer-enhanced robotic cardiac surgery: Experience in 148 patients," *The Journal of Thoracic and Cardiovascular Surgery*, vol. 121, no. 5, pp. 842–853, 2001.
- [5] S.-W. Kang, S. C. Lee, S. H. Lee, K. Y. Lee, J. J. Jeong, Y. S. Lee, K.-H. Nam, H. S. Chang, W. Y. Chung, and C. S. Park, "Robotic thyroid surgery using a gasless, transaxillary approach and the da Vinci s system: The operative outcomes of 338 consecutive patients," *Surgery*, vol. 146, no. 6, pp. 1048–1055, 2009.
- [6] P. Yohannes, P. Rotariu, P. Pinto, A. D. Smith, and B. R. Lee, "Comparison of robotic versus laparoscopic skills: is there a difference in the learning curve?" *Urology*, vol. 60, no. 1, pp. 39–45, 2002.
- [7] G. P. Moustris, S. C. Hiridis, K. M. Deliparaschos, and K. M. Konstantinidis, "Evolution of autonomous and semi-autonomous robotic surgical systems: a review of the literature," *The International Journal of Medical Robotics and Computer Assisted Surgery*, vol. 7, no. 4, pp. 375–392, 2011.
- [8] R. C. Jackson and M. C. Cavusoglu, "Modeling of needle-tissue interaction forces during surgical suturing," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2012.
- [9] —, "Needle path planning for autonomous robotic surgical suturing," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2013.
- [10] R. C. Jackson, R. Yuan, D.-L. Chow, W. Newman, and M. C. Cavusoglu, "Automatic initialization and dynamic tracking of surgical suture threads," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2015.
- [11] B. Kehoe, G. Kahn, J. Mahler, J.-H. Kim, A. Lee, A. Lee, K. Nakagawa, S. Patil, W. D. Boyd, P. Abbeel, and K. Goldberg, "Autonomous multilateral debridement with the raven surgical robot," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2014.
- [12] B. Kehoe, G. Kahn, J. Mahler, J.-H. Kim, A. L. A. Lee, K. Nakagawa, S. Patil, W. D. Boyd, P. Abbeel, and K. Goldberg, "Autonomous tumor localization and extraction: Palpation, incision, debridement and adhesive closure with the da vinci[®] research kit," in *Hamlyn Surgical Robotics Conference*, 2015.
- [13] T. Liu and M. C. Cavusoglu, "Needle grasp and entry port selection for automatic execution of suturing tasks in robotic minimally invasive surgery," *IEEE Transactions on Automation Science and Engineering*, vol. 13, no. 2, pp. 552–563, April 2016.
- [14] R. C. Jackson, V. Desai, J. P. Castillo, and M. C. Cavusoglu, "Needle-tissue interaction force state estimation for robotic surgical suturing," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct 2016, pp. 3659–3664.
- [15] F. Nageotte, C. Doignon, M. de Mathelin, P. Zanne, and L. Soler, "Circular needle and needle-holder localization for computer-aided suturing in laparoscopic surgery," vol. 5744, 2005, pp. 5744–5744.

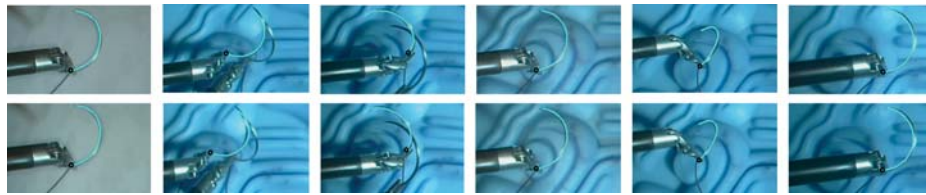


Fig. 11. The sequence of images correspond to the tracking algorithm results in different experiments. Top and bottom row images show the left and right camera images from the endoscope respectively.

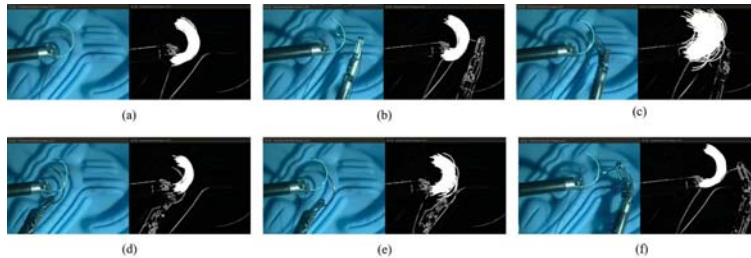


Fig. 12. The sequence of images (a-f) from tracking algorithm evaluation under occlusions using 300 particles. These images are captured from a single trial. The cyan lines mark the best matched particle overlaid on the endoscope left image view. In order to show the particle distribution, the segmented image of the left camera view is overlaid with all of the particles. During the tracking, high occlusions are created manually to test the performance of the proposed method. In (d), particles were briefly "distracted" since the occlusion was almost identical to the tracked needle. However, the tracking algorithm was able to recover from this distraction in the time steps immediately thereafter.

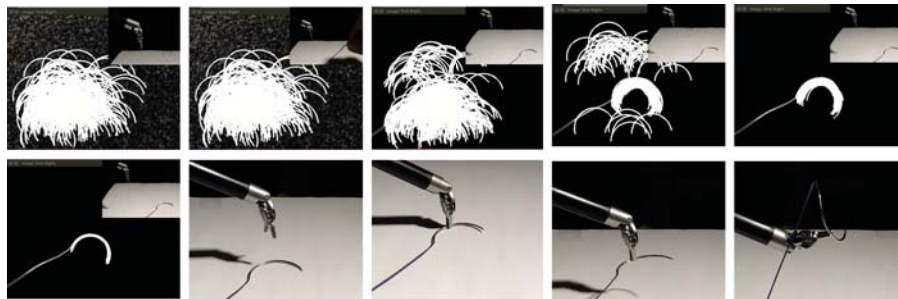


Fig. 13. The sequence of images correspond to the needle grasping demonstration. The particle filter algorithm was initialized with 300 particles and randomly sampled in the scene. Then the surgical needle was randomly inserted to the scene. The particle filter algorithm detected the needle and converged to a solution which accurately represented the position and orientation of the surgical needle. The robot manipulator was then commanded to approach to the location, perform grasping, lift the needle and move it about randomly.

- [16] S. Sen, A. Garg, D. V. Gealy, S. McKinley, Y. Jen, and K. Goldberg, "Automating multiple-throw multilateral surgical suturing with a mechanical needle guide and sequential convex optimization," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2015.
- [17] S. Iyer, T. Looi, and J. Drake, "A single arm, single camera system for automated suturing," in *Proceedings - IEEE International Conference on Robotics and Automation (ICRA)*, 05 2013, pp. 239–244.
- [18] 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," in *Proceedings - IEEE International Conference on Control, Automation and Systems*, 10 2013.
- [19] M. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking," *IEEE Transactions on signal processing*, vol. 50, no. 2, pp. 174–188, 2002.
- [20] C. Choi and H. Christensen, "Robust 3d visual tracking using particle filtering on the special euclidean group: A combined approach of keypoint and edge features," *The International Journal of Robotics Research*, vol. 31, no. 4, pp. 498–519, 2012.
- [21] M. Breitenstein, F. Reichlin, B. Leibe, E. Koller-Meier, and L. V. Gool, "Robust tracking-by-detection using a detector confidence particle filter," in *IEEE 12th International Conference on Computer Vision*, 2009.
- [22] S. Thrun, "Particle filters in robotics," in *Proceedings of Eighteenth Conference on Uncertainty in Artificial Intelligence*, 2002.
- [23] G. Kitagawa, "Monte carlo filter and smoother for non-gaussian nonlinear state space models," *Journal of Computational and Graphical Statistics*, vol. 5, no. 1, pp. 1–25, 1996.
- [24] C. Choi and H. Christensen, "Robust 3d visual tracking using particle filtering on the special euclidean group: A combined approach of keypoint and edge features," *The International Journal of Robotics Research*, vol. 31, no. 4, pp. 498–519, 2012.
- [25] S. Thrun, W. Burgard, and D. Fox, *Probabilistic robotics*. MIT press, 2005.
- [26] R. C. Jackson, R. Yuan, D. L. Chow, W. S. Newman, and M. C. avuoglu, "Real-time visual tracking of dynamic surgical suture threads," *IEEE Transactions on Automation Science and Engineering*, vol. PP, no. 99, pp. 1–13, 2017.
- [27] A. Frangi, W. Niessen, K. Vincken, and M. Viergever, *Multiscale vessel enhancement filtering*. Berlin, Heidelberg: Springer Berlin Heidelberg, 1998, pp. 130–137. [Online]. Available: <https://doi.org/10.1007/BFb0056195>
- [28] G. Steger, "An unbiased detector of curvilinear structures," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1998, pp. 113–125.
- [29] P. Kazanzides, Z. Chen, A. Deguet, G. S. Fischer, R. H. Taylor, and S. P. DiMaio, "An open-source research kit for the da vinci® surgical system," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2014.