
Robot Assisted Needle Placement: Developed Using Image Guided Surgery Toolkit (IGSTK)

Release 1.00

Peng Cheng¹, Andinet Enquobahrie², Roland Stenzel¹, Ralph Lin¹, Hui Zhang¹,
Ziv Yaniv¹, Hee-su Kim³, Kevin Cleary¹

July 10, 2006

¹Georgetown University, ISIS Center, Washington, DC 20007 USA

²Kitware Inc. Clifton Park, NY 12065, USA

³Department of Computer Science, Kyungpook National University, Korea

Abstract

Robotic systems for positioning needles or needle-shaped tools have the potential in increasing accuracy and speed of clinical procedures such as biopsy, radio frequency ablation, and target drug delivery. These systems can also minimize patient trauma. This paper describes a robot assisted needle placement system developed using the Image Guided Surgery Toolkit (IGSTK). IGSTK is an open source software toolkit that provides a robust and safe platform for researchers and clinicians to prototype image guided applications quickly and efficiently.

Contents

1. Introduction	2
2. Method	3
3. Implementation	5
4. Results	5
5. Discussion & Conclusion	7
6. Disclaimer & Acknowledgement	8
References	8

1. INTRODUCTION

1.1. Image-guided needle placement

Needles and needle-like tools are widely used in the clinical environment, especially in minimally invasive procedures for both diagnosis and treatment. One of the most common procedures is needle biopsy, in which clinicians deploy needles into a patient's body to sample a small amount of tissue for laboratory analysis. This procedure is used most often for tumor diagnosis and cancer staging in lung, breast, liver, and prostate. In another procedure known as RF ablation, surgeon inserts an ablation needle into the center of a tumor and uses the radio frequency energy to 'cook' the tumor. Both procedures are minimally invasive, but they also require a great amount of experience to accurately target the tumor. By overlaying real-time location of a tracked surgical tool on top of pre or intra-operative images, image-guided technology can allow clinicians to see patient anatomy, and thus can increase the accuracy of minimally invasive procedures.

1.2. Needle placement robot

This robot is designed and manufactured by ARC Seibersdorf Research GmbH, Austria [1]. (Figure 1 left). The robot has two joints (upper box and lower box, which can move parallel to each other) and 4 degree of freedoms ($\pm 19\text{mm}$ in translation and $\pm 30^\circ$ in rotation). Figure 1 right shows the system setup. The robot is mounted on the CT table after the patient is in place. Next the robot arm is adjusted to position the needle holder close to the biopsy area. Then a CT scan is acquired and loaded into the robot assisted needle biopsy application. Then the surgeon examines image slices, identifies tumors, and plans an optimal biopsy path that avoids important and vulnerable organs and tissues by setting proper target and entry points. The robot will then move the needle holder and align it with the planned path. Finally, the surgeon can advance the needle manually to hit the target. The robot can be operated remotely by multiple clients through TCP/IP communication. A remote client must connect to the server application as an active client before commanding the robot.

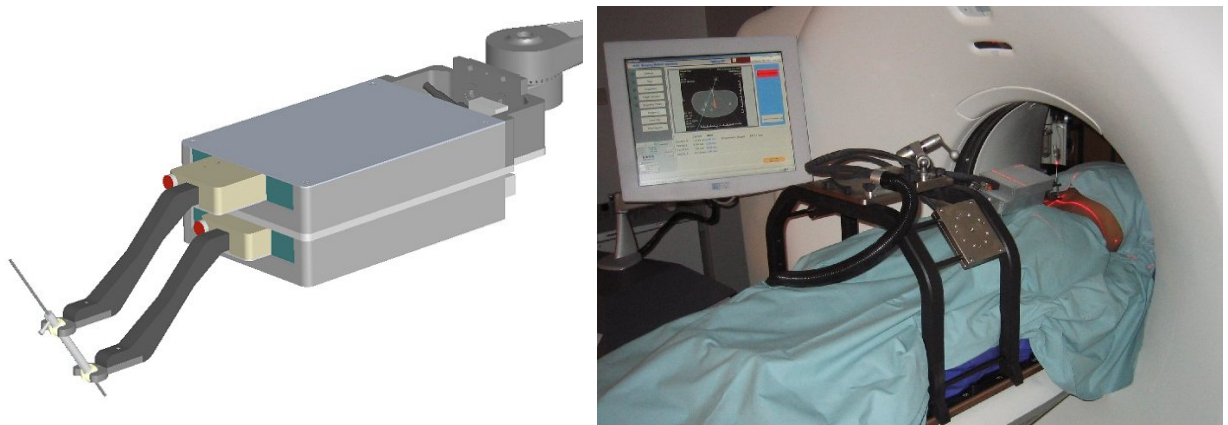


Figure 1. Deck of card robot (left) and robot assisted needle biopsy system setup (right)

1.3. Image-Guided Surgery Toolkit (IGSTK)

IGSTK is an open source toolkit designed for the development of image guided surgical applications [2, 3]. IGSTK has been developed on top of three other open source toolkits: ITK for segmentation and registration, VTK for visualization, and FLTK for graphical user interface. IGSTK contains the basic

components necessary for image-guided surgery applications, such as view classes for displaying and presenting results to the clinician; spatial object and spatial object representation classes for modeling and displaying physical objects including images and anatomical structures; and tracker classes for handling and communicating with tracked surgical tools.

2. METHOD

One critical step in a robot-guided needle placement application is to determine the transformation parameters between the robot and patient/image coordinate system. This registration procedure can be performed using a fiducial-based (point-based) registration technique. In this procedure, surgeons examine an image slice by slice, identifying fiducial markers and establishing the pairing to the physical fiducial markers. However, this manual tagging and matching procedure is time consuming and is prone to human errors. Hence, we developed an automatic fiducial marker detecting and matching algorithm.

We embedded 1mm-in-diameter metal fiducial markers (19 in total) onto the surface of the cylindrical needle holder (Figure 2 left). Figure 2 right shows a 3D reconstructed image of the fiducial markers, reconstructed from a CT scan. The markers follow a spiral pattern. The positions of each fiducial marker with respect to the robot space origin are known and are referred to as the model points later in this paper.

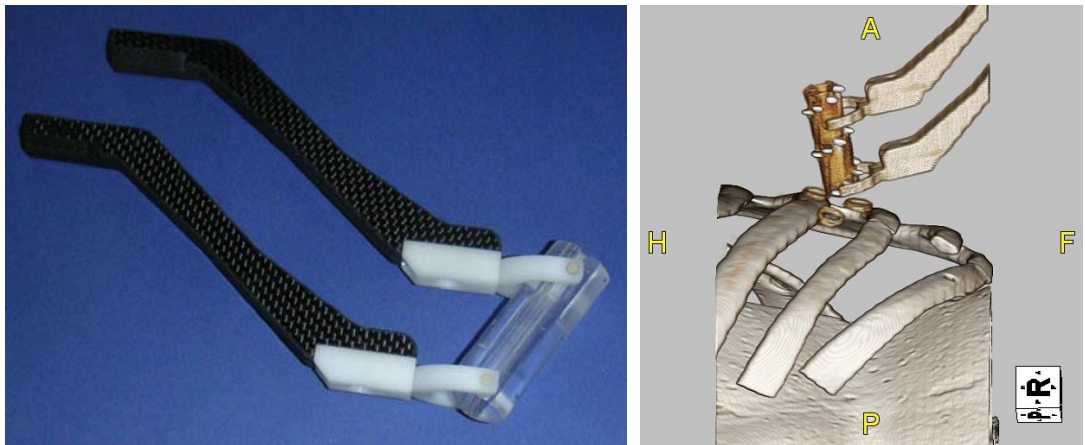


Figure 2. Needle holder with embedded fiducial markers (left), and reconstructed image showing the fiducial markers (right).

2.1. Fiducial point detection

Metal fiducial markers have a very high absorption rate in CT images. Thus we can threshold the image, extract high intensity objects, and then calculate the centroid of each object as a fiducial point position. In the case of the existence of other metal objects in the image, we can use maximum and minimum size criteria to filter out non-fiducial points such as the needle, the metal part of the robot, and any metal implantation inside a patient.

However, Even if we use a size restriction, there will still be some small objects in the segmentation results that can not be separated from the true fiducial points. Thus, the goal for fiducial clustering is to filter out those false positive segmentation results. Given the high density distribution of fiducial markers in the end effector (19 fiducials in total), and as long as no metal object except fiducial points are detected

within or around the plastic needle holder (the needle itself can be filtered out by size criteria), we can conclude that the remaining false positives are all outliers. We can then calculate the distance map for the segmented point set, and iteratively delete point with the largest average distance to the rest of the points, until the maximum distance between point pair is lesser than or equal to the maximum scale of the model. This method is similar to the max-cut clustering algorithm. A more sophisticated algorithm can be used to cluster the similarity map between the segmented points and the model points.

However, this technique will not necessarily segment out all 19 of the fiducial points that are required for the next step, landmark-based registration. If a high threshold value is used, then we might misdetect some of the fiducial markers. If a low threshold value is used, then we might end up detecting too many false positive points, and thus, we would not meet the assumption for clustering algorithm, that no points other than fiducial markers are detected within or around the needle holder area. To address this issue, we used a two-stage segmentation algorithm (Figure 3). First we use a high threshold value (3000 Hounsfield Unit) to segment the whole volume, followed by a clustering algorithm to extract the high confidence fiducial markers. This step will guarantee that all the resulting points are true positives. Then, we use these points to extract the volume of interest, which contains only the needle holder. This time, we perform the segmentation again on the sub-volume with a lower threshold (2000 Hounsfield Unit), followed by the clustering. This step typically results in segmenting all the fiducial points. Because the second step is applied to a sub volume contains only the needle holder, it is much faster, and it will not generate false positive segmentation results.

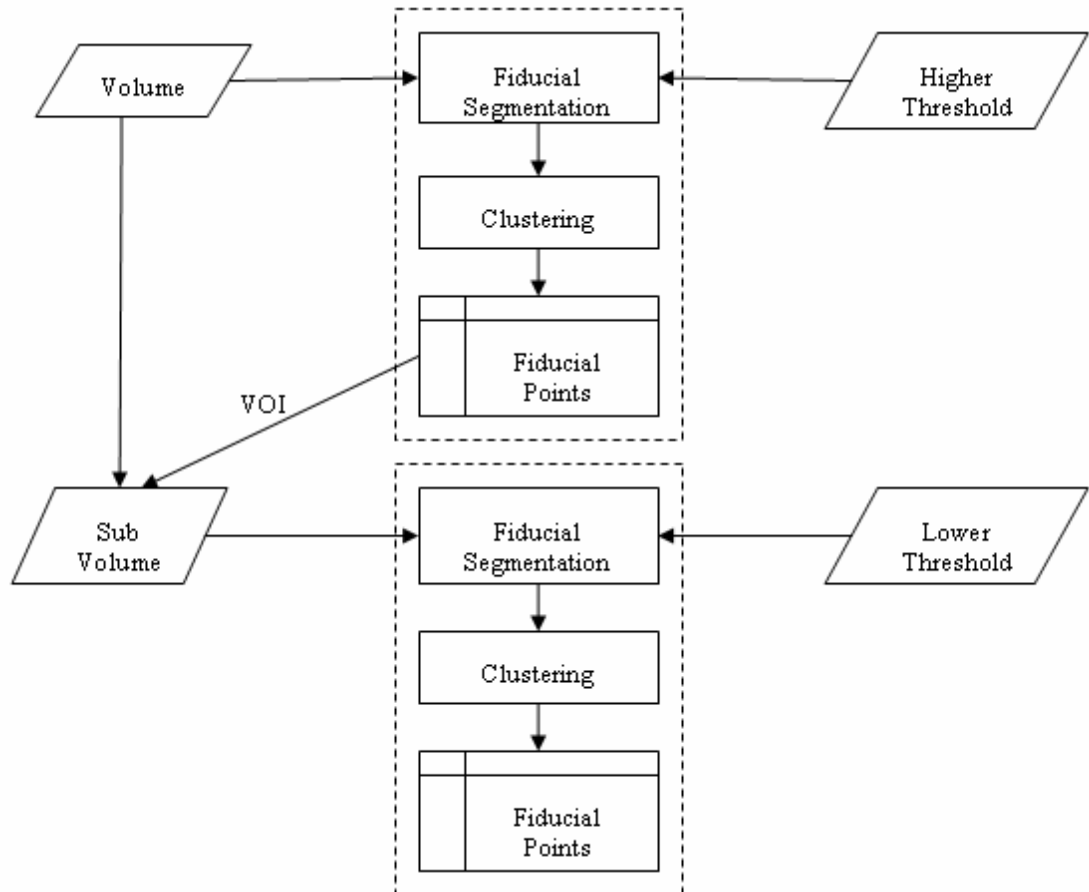


Figure 3. Flow chart for the two stage segmentation algorithm

2.2. Fiducial point matching and registration

After segmenting all the fiducial points in the image, we can then sort the points by projecting them on to their center axis, which is the axis of the cylinder-shaped needle holder. This axis can be approximated by extracting the principle component axis of the 3D points set. One design flaw of this particular end effector is that it is symmetric with respect to the center plane that is perpendicular to end effector's axis, making it hard for a computer program to identify the correct order of the point sequence. In this application we assume that the point with largest Y coordinate (the lowest elevation, because the robot can not flip upside down, so the end effector is always pointing downwards) corresponds to the smallest Z axis point in the model. A better solution to this problem is to manually take out one of the fiducial points on either the upper or lower side of end effector and thus to make the fiducial model asymmetric; then to try the landmark-based registration with two different orders of the points; and finally, to take the registration result with the smaller RMS error.

3. IMPLEMENTATION

The fiducial segmentation and matching algorithm was implemented using ITK classes. The algorithm takes an ITK image, a threshold, and maximum and minimum sizes as input, and then generates a segmented points list. The major classes used are [4]:

[itk::BinaryThresholdImageFilter](#)

[itk::ConnectedComponentImageFilter](#)

[itk::RelabelComponentImageFilter](#)

The fiducial clustering filter takes a list of sample points and list of the model points as input, and then returns the clustered points list. The communication between the application and robot server is through the TCP/IP protocol. For this purpose, the socket communication component of IGSTK was used. Other modules of this application, such as user interface, visualization, registration and path planning are implemented using the view, and registration components of IGSTK.

4. RESULTS

4.1. Application GUI

Figure 5 shows the user interface of the application with control panel on the left, 3 standard 2D slice views and a 3D volume rendering on the right. The yellow cylinder is the needle holder, and the purple square indicates robot's working region, in these images, the robot is being aligned along a path that has been planned to targeting tumor while avoiding ribs.

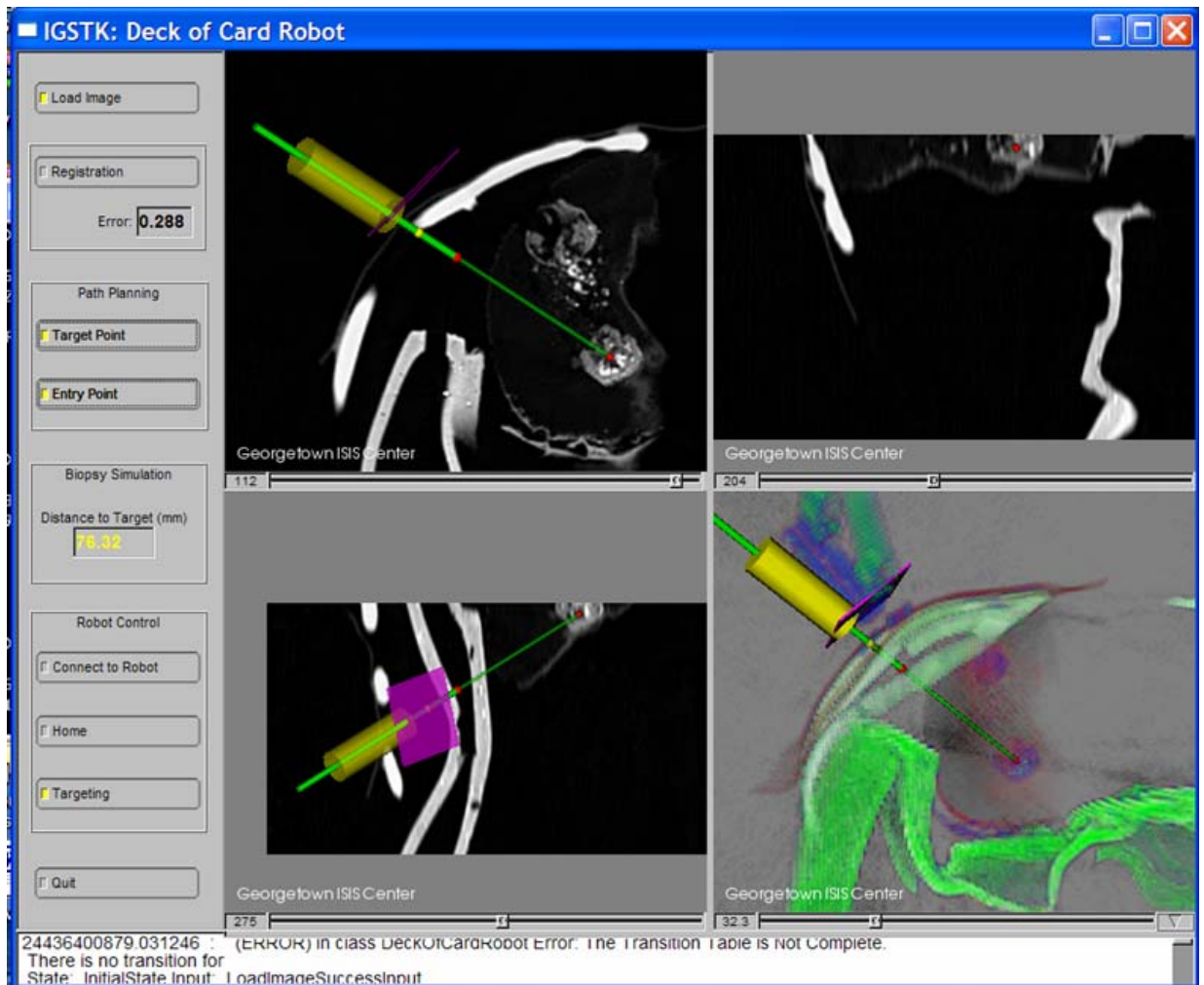


Figure 4. Application user interface showing the registered robot

Figure 5 shows the phantom study setup. After registering the robot, the application can command the robot to align with the planned path, and the robot successfully targets the pre-attached skin fiducial markers along the planned path.



Figure 5. Phantom study for the needle placement robot

4.2. Validation study

To validate the whole system - especially the robot registration and the communication between the application and the robot - we acquired three groups of data. Each group contained one home position and three other know robot positions for 12 scans in total. The image was reconstructed with $0.637 \times 0.637 \times 1.00 \text{mm}$ resolution. It took about 30 seconds to run the algorithm on a moderate PC with an image size of $512 \times 512 \times 105$. The automated algorithm can successfully registered all 12 data sets with an accuracy of $0.298 \pm 0.018 \text{mm}$ RMS.

5. DISCUSSION & CONCLUSION

The results showed that our registration method is robust. However, there are a number of areas for improvement.

We are currently investigating more relaxed registration methods such as iterative closest points and model-to-image registration which do not require the exact same number of segmented fiducial points as in the model and the points pairing, and compare their success rate and accuracy to the this landmark based method.

Furthermore, our algorithm does not take respiratory motion effect into consideration. In lung and liver biopsies, respiratory motion can introduce errors as much as 5 cm into the system. Therefore, as we continue to develop this application, we may need to employ motion compensation techniques to reduce the motion error.

6. DISCLAIMER & ACKNOWLEDGEMENT

This application is being developed as an example application with the release of IGSTK open source software toolkit, and it is not suitable for clinical use.

The author would like to thank for help from the other IGSTK developers while developing the system, there are Luis Ibanez, Julien Jomier, Stephen Aylward, Rick Avila, David Gobbi, Brain Blake, and Kevin Gary. The robot system was designed and constructed by Gernot Kronreif of ARC Seibersdorf Research. This research is supported by the National Institute of Biomedical Imaging and Bioengineering (NIBIB) at the National Institutes of Health (NIH) under grant R42EB000374 and by U.S. Army grant W81XWH-04-1-0078

REFERENCES

- [1] K. Cleary, A. Melzer, V. Watson, G. Kronreif, D. Stoianovici. *Interventional robotic systems: Applications and technology state-of-the-art*. Minimally Invasive Therapy. 2006; 15:2; 101-113.
- [2] P. Cheng. *IGSTK: Framework and Example Application Using an Open Source Toolkit for Image-Guided Surgery Applications*. SPIE Medical Imaging, February 2006.
- [3] K. Gary. *IGSTK: An Open Source Software Toolkit for Image Guided Surgery*. IEEE Computer, April 2006.
- [4] Ibanez, L., Schroeder, W., “*The ITK Software Guide*”, ISBN 1-930934-10-6, 2005. <http://www.itk.org/ItkSoftwareGuide.pdf>