

Mixed Reality Surgical Team Training: Endoscope surgery simulator

*Computer Integrated Surgery II
Spring, 2022*

Haochen Wei, under the auspices of Professor Peter Kazanzides and Dayeon Kim

Introduction

- We use the Augments reality technic to create a simulation environment to train the surgeon. The simulator will locate the position and pose of the tools and will overlay tools on the virtual anatomy structure. We created a demo that can achieve the object but planning to further develop. The final goal is that system can be integrated with the Da Vinci System build-in simulator to train the whole team together.
- System will solve the problem that currently Da Vinci simulator only covered the machine side surgeon, and the system is capable to suit the general endoscopic surgeon training.

Current Problems

- Laparoscopic surgery has many advantages over traditional open surgery. But Due to the difficulty, it required lots of training from a trainee to a qualified surgeon. Recently several Augmented Reality based surgical simulator has been developed, but most of them only considered several specific types of procedure and ignore the robot-assist surgery system integration.
- Because Da Vinci also required some assistants to hold tools near the bed, and their technic is like regular endoscopic surgery, an AR simulator that suits Da Vinci is necessary at this moment.

Technical Approach

Obtain position and pose of the endoscopic tools:

1. Detect raw feature points.

Using binary method to detect the dark area of the image. Center pixel of the gray area as a raw feature point.

2. Filter the feature points using geometry constrain

Use straight-line detection algorism was used to initially filter out the noise that came from the dark area of the image.

3. Pair feature points that stay at the same height.

Obtain the space coordinate of the feature point using the triangulation method. Calculate 3d space line using least square method.

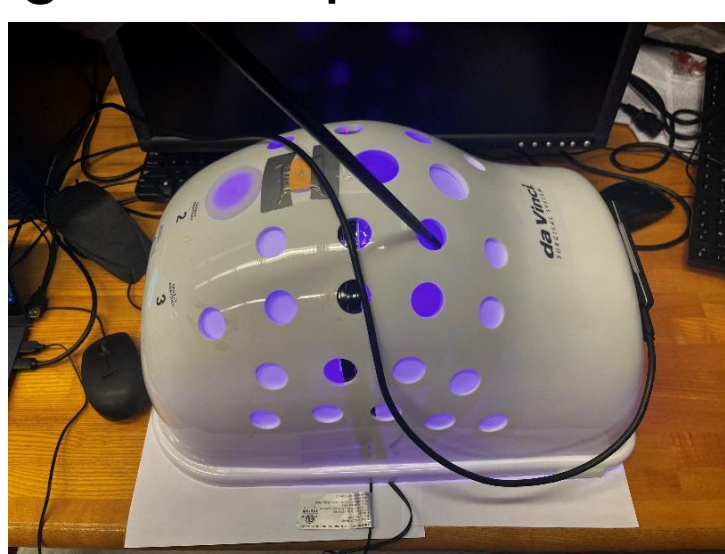


Figure 1: Hardware setup: outside

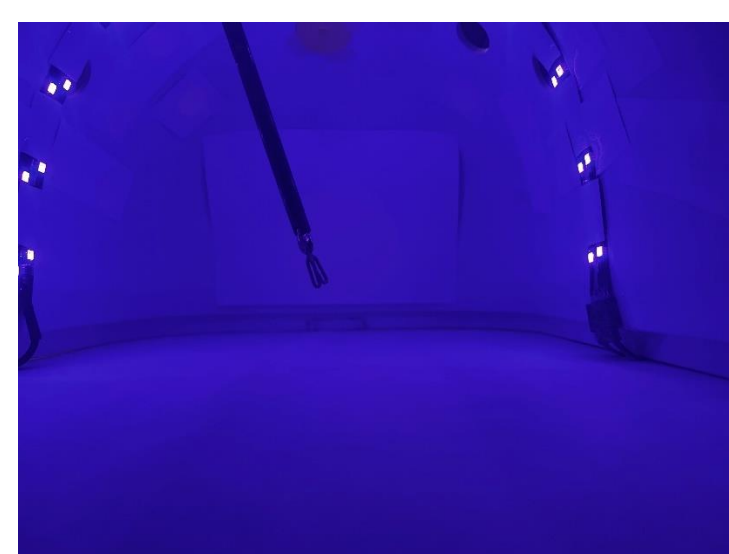


Figure 2: Hardware setup: outside

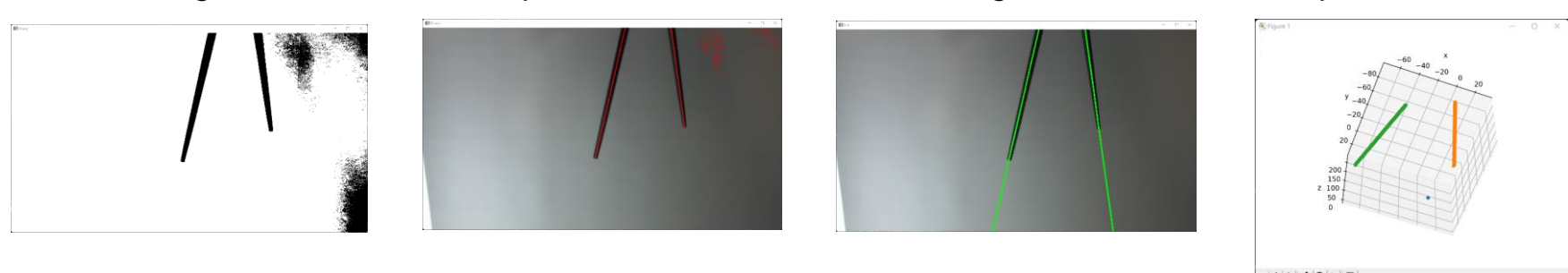


Figure 3-6: Pipeline of obtaining feature points

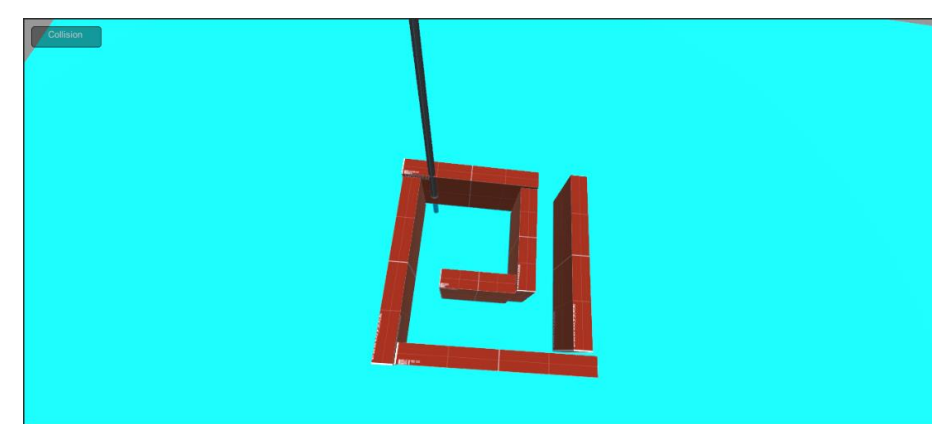


Figure 7: UI: Collision

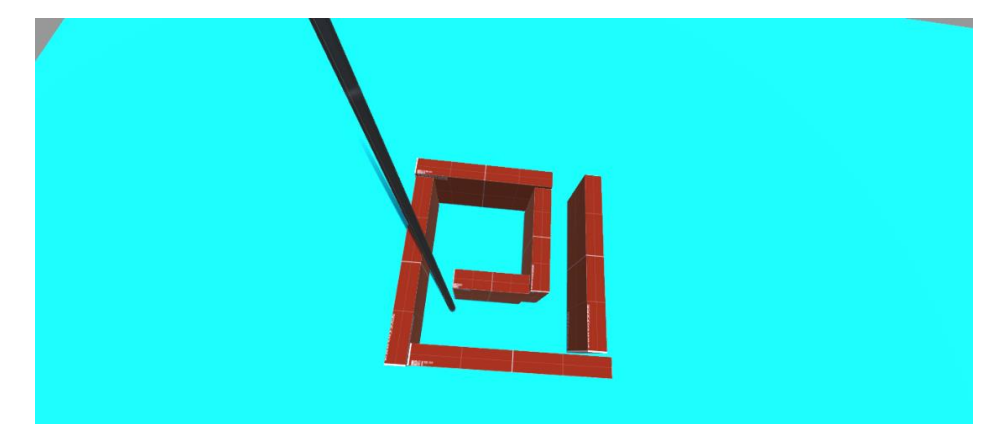


Figure 8: UI: No collision

UI that display the object:

Unity as platform. Using Cylinder game objects to mimic endoscopic tools. Using unity build-in collision detection function. Training environment was built within the unity engine.

Results and Problems

Performance:

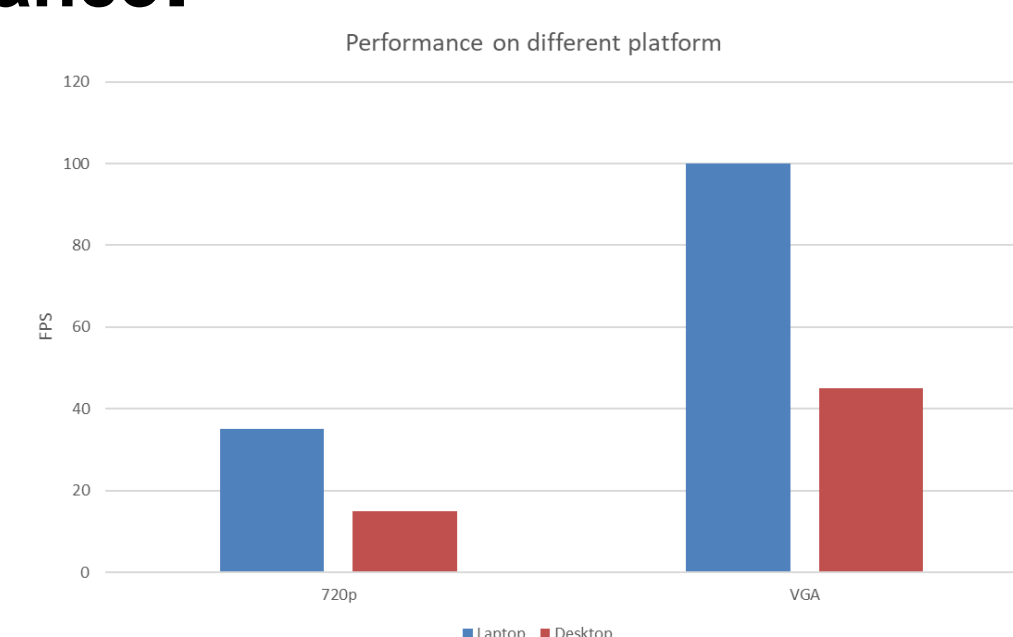


Chart 1: Performance on different platform and resolution

Note: CPU Clock on Desktop: 4Ghz Laptop:1.5Ghz

Accuracy:

Reprojecting the 3d point back to the image to measure the error. At the distance of 20cm from the lens to the object, 720P image, Average pixel difference is 1-2. The distance error is 0.06 cm-0.12 cm (angular difference 0.003-0.006rad).

Problems:

1. Real-time usage will encounter fluctuation.
2. Localization along tool direction is relatively poor
3. Tools registration.

Future Work

- All the problem addressed in the "Problem" section
- Identify the type of tools
- Using the point cloud to reconstruct identified tools.
- Combine the simulator with the Da Vinci Robot.
- User study.

Lessons Learned

- Some minor problems can stop going for a long time.
- Make dependencies in the plan.

Publications

- Being prepared to submit as a conference paper once future works are finished

Support by and Acknowledgements

- Thanks to JHU LCSR, Prof. Peter Kazanzides, and Dayeon Kim.

