Automating Camera Movement based on Gaze tracking in Surgical Robotics

Olivia Raisebeck
Computer Science Department
Worcester Polytechnic Institute
Worcester, MA, USA
oeraisbeck@wpi.edu

Soham Aserkar
Robotics Engineering Department
Worcester Polytechnic Institute
Worcester, MA, USA
ssaserkar@wpi.edu

Uthiralakshmi Sivaraman
Robotics Engineering Department
Worcester Polytechnic Institute
Worcester, MA, USA
usivaraman@wpi.edu

***Abstract*—The advent of robotics in surgery has improved the accuracy and efficiency of robot assisted minimally invasive surgeries (RAMIS). Specifically, with the introduction of vision sensors and vision based algorithms, results have improved the in-depth perception and made it possible to have multiple point of views, both first and third person. However, there are several challenges as well. Specifically, in tele-operated robot surgery, it is challenging to operate both the camera as well as surgical tools simultaneously. The situation results in increased cognitive workload for the surgeon. A vital part of the surgery as not having proper vision access or viewpoint can be detrimental in capturing all the tools present in scene and thus impact the patient safety and overall surgical accuracy.**

To tackle the problems, we describe a gaze tracking based automation of endoscope camera of the Da Vinci Research Kit for Robot assisted minimally invasive surgeries and semantic segmentation to identify various region and do a region of interest mapping . We also use Tobii Eye Tracker 5 for detecting Gaze tracking parameters. We compare the gaze based automation with two different interfaces, keyboard and mouse and do a user study based comparison and evaluation of different interfaces to obtain subjective feedback from users.

***Index Terms*—Endoscope Camera Automation, Da Vinci Research Kit, Tobii Eye Tracker 5, Unity**

I. Introduction

Surgical robotics has already come a long way in the past decades. In the present situation, most robots have been tried and tested to have one or more capabilities that are needed in the Operating Room/Theatre, yet there are many challenges to be addressed. One of the main objective of introducing robots is to reduce surgeons workload and help automate the surgical process partially or fully.

A typical RAMIS system is a teleoperative system where the surgeon operates a couple of robotic masters while seated at a console to position and move the surgical instruments inside the patient, such as the most popular RAMIS system, the da Vinci surgical system by Intuitive Surgical Inc., Sunnyvale, California. Such an approach brings about what accounts as success for both the surgeon and the patient. On the one hand, decreased blood loss and pain shorten hospital stays and speed up recovery; on the other hand, motion scaling, tremor filtering, better workstation ergonomics, and fulcrum effect compensation improve the surgeon's dexterity and lessen his weariness. The improved vision and the return of depth perception are two additional important benefits offered by RAMIS. In actuality, as surgical techniques advanced, the viewing modalities underwent significant modification. Robotic surgeries can help reduce fatigue and stress, minimize effort and maximize outcome. In that as well, specifically we would like to optimize the detection of tissues, track gaze to position the endoscope camera and in the long run, help the robot learn the surgical procedure. According to the surgeon's instructions, an assistant in LMIS is in charge of supporting and positioning the endoscope. This feature necessitates precise coordination between the two, which is frequently challenging to achieve. Robotic-Assisted Endoscopic Manipulators (RAEMs) have been created to return the direct control of the surgeon's field of view to them. In RAMIS, the endoscope and surgical instruments are both directly teleoperated by the surgeon. This strategy exposes two primary issues. First off, switching between the camera and the instruments repeatedly makes RAMIS operations less fluid and increases operating times. Second, dealing with the new endoscope control dynamics complicates camera motion and adds to the surgeon's cognitive workload, which might result in mistakes when executing surgical operations.

An eye tracker-based robot control system is helpful. The system's ability to lessen user weariness induced by the operator's wearing sensors or maintaining attitude. Decrease Surgeons' mental workload using gaze tracking to automate camera position than using foot pedal.

In this work, we propose to enhance the surgical environment by optimizing the detection of tissues, tracking gaze to position the endoscope camera in order to get better viewpoints in the surgical region of interest. We also do a comparative study between gaze and other interfaces to evaluate subjective measure and feedback from users.

There are seven sections to this report. A quick overview of the current techniques used in automating camera viewpoint in described in Section I. The studies of previous researchers in the field that served as inspiration for this one are described in Section II. The tools, software, and environments used in this work are described in Section III. The methods used to automate the camera motion and get surgical region of interest are outlined IV. The methods employed are briefly described in Section V. Finally, Sections VI describe about the results and conclusion obtained. Section VII address the key outcomes and future work.

II. Related Work

There have been many attempts at trying to automate camera motion in many scenarios, ranging from automating multiple views in games, improving multiple viewpoints in surgeries. In this section, we briefly describe about the various work and attempts that focused on automating camera motion in different scenarios.

In [1], eye tracking control of manipulators is analysed. Manual movement of robots can cause fatigue and lots of effort by the operator when used in high stress environments such as surgery or for rescue missions. Using eye tracking would be a way for these operators to control the robots in an almost subconscious way to make it easier on them. All the experiments test how quickly the operator completes the task using a joystick compared to gaze tracking, at various speeds of the robots. In this paper the following method is proposed. The screen is divided into a 3x3 grid. When the gaze is in the center, the robot will not move; at any other point on the screen, the robot will move until the gaze is in the center. The robot moves with a predetermined constant speed that is varied during different runs of the experiment. There are three experiments that will be run. The first involves a laparoscope holder robot, which uses EMARO and has three rotational degrees of freedom and one transnational degree of freedom. The participant moves the camera until the desired object is in the camera view, once with a joystick and once with their gaze. The second is driving a crawler type robot where the goal is to move the robot with their gaze or the joystick to a desired point. The last uses a power shovel with a tele-operated camera control system.

In [2], A method to analyse the operator gaze for wrist mechanism for surgical robot for suturing task is described. Suturing can be difficult, especially with pediatric surgery as the needle view can often be obscured by tissue or other obstacles and not be visible. While many surgery robots currently exist, most have a master/slave relationship where the person is completely in control of all the actions of the robot. For suturing this may not be ideal since the needle is not always in sight. Many studies have shown a connection between viewpoint and cognition, so instruments partially controlled by the surgeon's gaze could be helpful in accomplishing this. The goal is to use viewpoint trajectory analysis to inform surgical robot design. This experiment analyzed where the participants were looking in order to map of the gaze to create an algorithm of how to predict suturing motions. An issue that arises in the calculations was that the best wrist length was too large to be used in small suturing situations such as for children.

In [3], a method to understand the impact of visual- spatial ability on laparoscopic camera navigation training is analysed. Proper assessment of visual spatial ability while manipulating objects is a hard problem statement. Surgeons and camera assistants are faced with technical difficulties during minimally invasive surgery (MIS). They must cope with DOF, fixed access points, among other things. a monocentric, prospective trial. Good camera navigation is defined by the preference of the individual surgeon and it includes an anticipatory knowledge of the surgical procedure as well as the skill of translating movements from a 2D-vision into a 3D-environment.

The paper [4], plays a certainly important role to have the metric and ability to evaluate the camera assistant skills to operate the endoscope camera. The authors have developed a clinically feasible rating scale for structured assessment of the same. The authors defined the following key aspects of LCN: operational field centering, correct angle of the horizon, correct instrument visualization, verbal commands from the operating surgeon, and manual corrections from the operating surgeon. As to how the current skill assessment will be affected is a little subjective as some are good at navigating through certain areas than others and there can be no global metric to look at the skill in question.

In [5], a system to analyse the camera semi-autonomous and autonomous navigation in robotic- assisted surgery is analysed. The research is centered on getting at a shared autonomy, which means that the surgeons will never be kept in the dark about the decisions made by the autonomous endoscope camera movement. This enables the surgeon to take well informed decisions in real time about the patient and is able to complete the task at hand with a reduced workload and the fear of NOT knowing what the robot may or may not do.

However, in certain cases the surgeon will have some level of doubt, and may want to switch to manual control altogether, which has been provided. Manual override is possible at anytime and it is only fair, as not always the 'trust' factor will be present, and moreover, it is vital that the patient is put first, always. The system was developed using a Virtual Reality framework in order to control the endoscope camera of the dVRK. The three modalities included in the workflow were Manual, Semi-autonomous and autonomous. Manual meant that the teleoperation will work as is, from the original dVRK that has a foot pedal to control the endoscopic camera module. Semi-Autonomous meant that the technique was to either use a foot pedal or directly control the robot like the manual camera modality described earlier. The implementation of the autonomous control modality is shown in the picture, that shows the scene center as the midpoint of the center of masses of both tools. The SCAN also included implementation of some types of zooming techniques used.

The key highlight of work in [6], is that it emphasises on the need for automating camera navigation in robot assisted surgery, proposes a method to do so and as well as provides insights into comparative study between manual and autonomous camera automation. Various attempts have been made to automate camera. Some of which are based on positioning the camera such that it lies on the centre of all the surgical tools or follows the tracking tools in the surgery and some other methods focus on estimating the best possible view and then positioning the camera based on some prediction or exploration algorithm.

The main work highlighted in [7], the authors propose a novel method for phase recognition that uses a convolutional

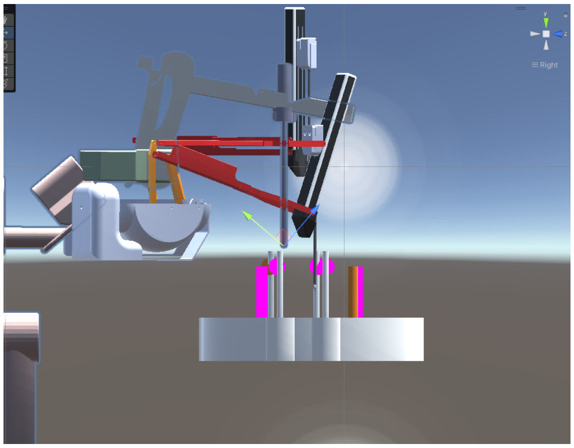


Fig. 1. Da Vinci Research Kit Setup in Unity Environment - View 1

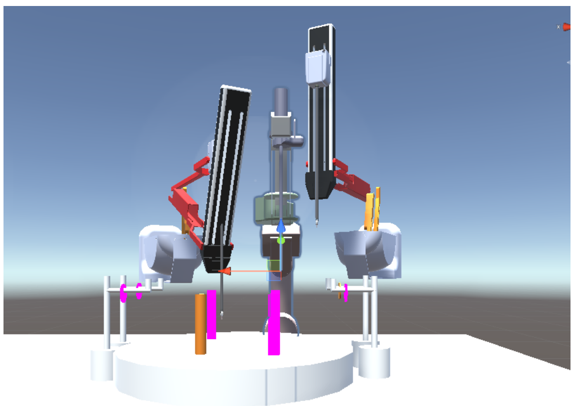


Fig. 2. Da Vinci Research Kit Setup in Unity Environment - View 2

neural network (CNN) to automatically learn features from cholecystectomy videos and that relies uniquely on visual information. It has been shown that the tool usage signals can provide valuable information in performing the phase recognition task. The authors present a novel CNN architecture, called EndoNet, that is designed to carry out the phase recognition and tool presence detection tasks in a multi-task manner. This was the first paper to ever attempt the recognition of the phase of the task in a surgical workflow.

III. Environment Setup and System Description

We used Unity simulation environment for simulating the Da Vinci research kit consisting of three manipulators, 2 PSMs (Patient Side Manipulators), ECM (Endoscope Camera movement), is a research platforms that explore innovative new concepts in minimally-invasive surgery and Unity Tobii Eye Tracker 5 with Software Development Kit, Unity SDK. We are using Unity version 2021.3.12f1 for this project and used C# programming language.

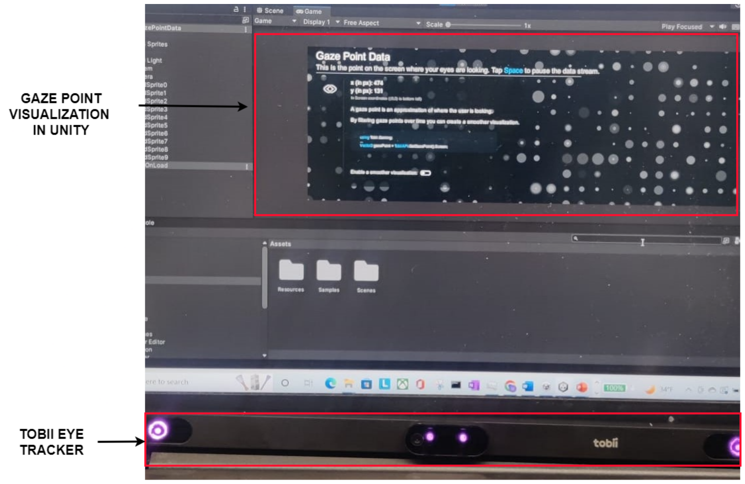


Fig. 3. Tobii Eye Tracker 5 with Unity SDK

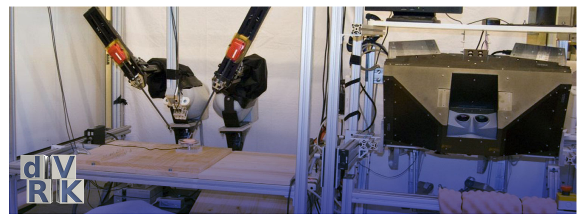


Fig. 4. Da Vinci Research Kitl

IV. Methodology

A. Camera Automation

We are using three types of Interface to move the camera. First we have the two that are closer to the current standards, by mouse and by keyboard. Then we have our method of using gaze tracking to move the camera. We want to judge these methods on two metrics: effort and how intuitive the system is. Gaze tracking would seem to be the most intuitive since you glance where you want to camera to go without even thinking about it. The next most intuitive we would guess is the mouse as we use mouses to scroll on our computer everyday and they were designed to be intuitive. Last we would say pressing keys are the least intuitive as unless someone plays video games often, we do not use keys for scrolling very often. For the

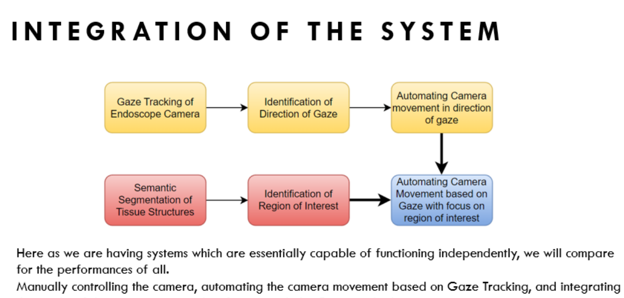
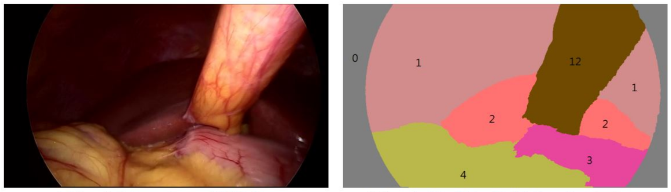


Fig. 5. Overall System Description



(a) Example Endoscope Image of Gallbladder

(b) Example Color Mask of Gallbladder and nearby tissues

Fig. 6. Example from the CholecSeg8k

$$Accuracy = \frac{T_p + T_n}{T_p + T_n + F_p + F_n}$$

$$Precision = \frac{T_p}{T_p + F_p}$$

$$Recall = \frac{T_p}{T_p + T_n}$$

$$F_1 = 2 \cdot \frac{precision \cdot recall}{precision + recall}$$

Fig. 7. Formulae for calculating Accuracy, Precision, Recall, F

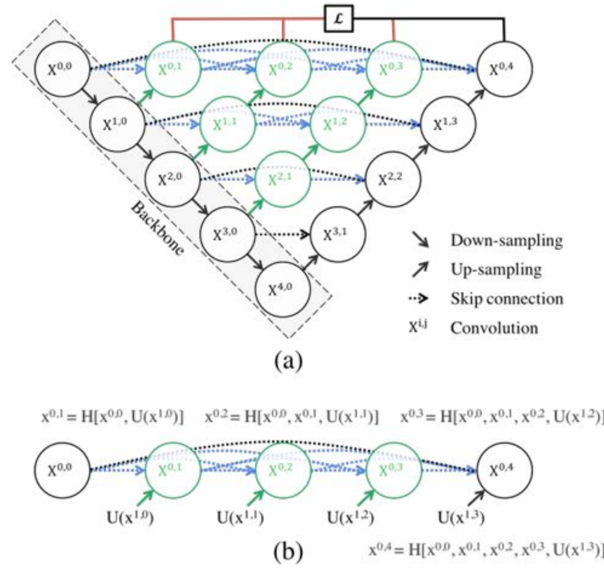


Fig. 8. Unet++ Network Architecture

next metric of how fatigued the user feels we also predict gaze tracking will score the best on. This is because it is the only one of the three methods that does not require movement from the user. Scrolling a mouse typically takes more effort than pressing a key so we expect the keys to score second and the mouse to scroll last.

B. Semantic Segmentation for CholecSeg8k

Based on Cholec80.We are working with the CholecSeg8k dataset for semantic segmentation. The dataset consists of 8080 images in total captured from various frames from the original Cholec dataset which consists of 26 videos. The dataset was originally used to identify the surgical workflow and detect the usages of the surgical tools as and when the need arises. It was also used for phase detection of the surgical workflow. In order to set the dataset which was obtained from Kaggle[]. The original dataset was made from videos about the surgical procedure called "Laparoscopic Cholecystectomy". While generating the CholecSeg8k dataset, few frames were ignored to ensure that the the dataset was feature rich and consisted of meaningful images.

We plan to use the UNet++ architecture for the training of the model. UNet is best known for its ability to perform medical image segmentation. UNet++ is an advanced version of the same. UNet++ differs from Unet in the form that in UNet the feature maps are directly received by the decoder, whereas in UNet++ the feature ,apps undergo a dense convolution block whose number of convolution layers depend on the pyramid level. Essentially, the semantic level expected in the decoder is received at a closer level in the encoder itself.

As for the training itself, we are using the cluster available at WPI to train our model and save the best weights that got the highest precision, recall and the F1 score. Additionally, we are monitoring the Dice Coefficient and the Intersection over Union(IoU). We are using the Adam optimizer with a learning rate of 3e-4. We have imported the resnet18 as weights for the model encoder so as to save time from training it again. The architecture of the network is described in Figure 8.

C. Region of Interest Classification

Writing Code for identification of region of interest required knowing where the segmentation map for the Gall Bladder would come. Hence, exporting the feature maps from the trained network, to only identify the Gall Bladder helped narrow down the region of interest.

V. Experiments

A. Automating Camera based on Gaze

1) Initial Proof of Concept: To first test of the proof on concept for gaze tracking we used a basic 1080p web cam and python code to report which cell of the 3x3 grid on the screen the user was looking at. It would track the position of the pupil by turning the current frame from the camera into a gray scale and use dlib's 68 point trained facial landmark detector to detect where the eyes are. From that, it would search for the center of the darkest part, which is the center of the pupil. Having the outer corners of the eye from the dlib library, we measure where the center of the pupil is on a scale from 0 to 1, 0 being all the way to the left or top of the eye and 1 being all the way to the right or bottom of the eye. Calibration was done by place a white dot at each point that represented the grid line of a third of the screen and having the user click the enter key when staring at that dot. The code would store the number, a decimal between 0 and 1, of what position in the eye the pupil was at in that moment.

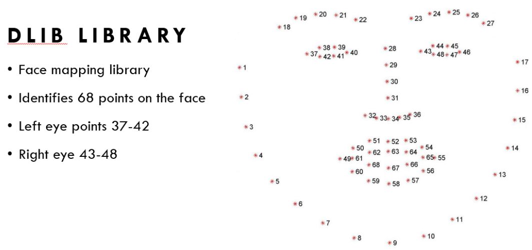


Fig. 9. Gaze Estimation based on DLIB

This was done four times to get the four grid line points of which when the gaze was beyond those points, it would print out that the gaze was that direction. Left and right were tracked well but the code had a harder time telling up and down, likely from there being a smaller difference between these two and the camera not being as high quality as one needed for very precise detection.

2) Experiment with Tobii Eye Tracker 5: We then moved to using the Tobii Eye Tracker 5 to measure our gaze tracking. After calibration, the Tobii Eye Tracker tells us what points on the computer screen the user is gazing at. Similar to the proof of concept, we split the screen into thirds. For our purposes, we put the coordinates right into the code but to be applied to other systems, they should be calibrated for that screen. We used the Unity simulation environment to simulate the camera and its movements. Instead of printing to the screen which way the gaze is looking, we move the camera with that information. We then set the grid so that when the gaze is anywhere other than the central cell, the simulated camera will move in that direction until the gaze is in the central cell of the grid. The Tobii Eye Tracker showed a lot more accuracy than the web camera especially with detecting upward and downward glances.

B. Identifying Surgical Region of Interest

We attempted to identify the Region of Interest based on the Semantically Segmented Image. Based on the pixel values obtained for the semantic segmentation, we were able to identify the map/class that we identified the 'desired organ of interest' in the map of the file. Accordingly, we drew a bounding box around it. Upon receiving this, we were able to copy the same coordinates in the original endoscope image as shown to the right in the case of the gall bladder.

C. User Study

To perform our study, we had thirteen participants to perform five tasks using our system. The first task was moving the camera to four cylinders in the simulation using the keys. The second and third tasks were the same but using the mouse and gaze respectively. The camera was considered on the cylinders when the dot in the center of the screen was on any point in the cylinder. We monitored the tests and provided verbal feedback to the participants of when they had completed the task. The fourth test involved a colored image on a plane in the simulation. The monitor would give a color to the user and the user had to get the central dot of the camera onto that color in the image. The monitors told the participants when they had succeeded. The last test had a box over the colored image. This box was representative of the region of interest. When the monitor confirmed the dot was within that box, the test was over. We then sent a survey to each user asking to quantify, on a scale of one to five, how much effort was required for each task and how intuitive each task was.

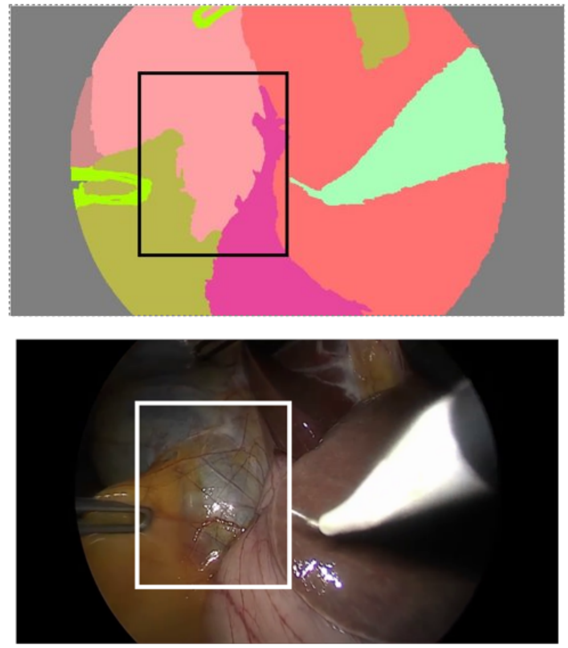


Fig. 10. Surgical Region of Interest Mapping/Tracking

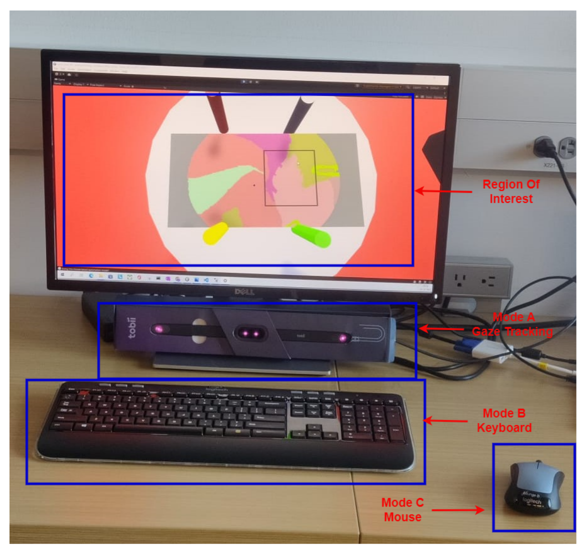


Fig. 11. User Study Setup

After our first two volunteers, we modified the experiment.

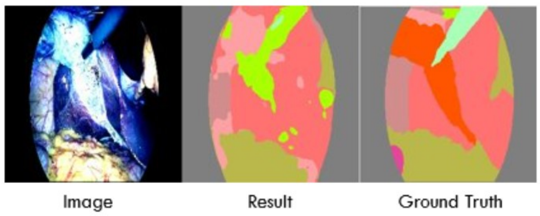


Fig. 12. Semantic Segmentation Results

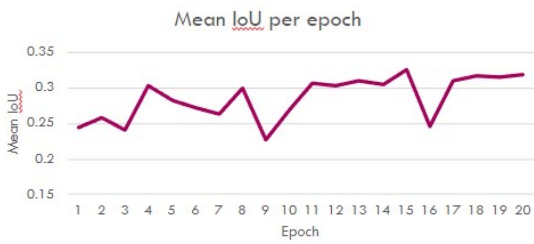


Fig. 13. Mean IOU per epoch for the Semantic Segmentation

The gaze tracking did not seem to work with the users. They had to look very dramatically in some directions in order to move the gaze and when looking in the center of the screen, the camera would move downwards. We believed this to be a calibration issue and for the next users re-calibrated both the Tobii gaze tracker and the same to the user before beginning the tests. The gaze tracker was calibrated with its standard calibration test involving the user look at 7 different points on the screen. The game was calibrated from determining where in the game we want the camera to move. Four points of gaze on an x,y scale based on the Tobii calculation were taken. These points were the points at which the camera would begin to move should the users gaze be outside of the square those points create, the camera would move until the gaze was in the frame. This allowed for a lot of further obstacles in the experiment. While the calibration was an improvement from the first two there still seemed to be more calibration issues than when we would test the software, likely because we had calibrated the game to our self multiple times and had familiarity with the system.

VI. Results

A. Semantic Segmentation Results

We trained the model on NVIDIA A100-PCIE-40GB MIG 1g.5gb, with a batch size of 2 for the training set, and 1 for the validation set. Further, we ran it for 20 epochs. After the training, we were able to achieve the following results: Mean Precision = 0.5408 Mean Recall = 0.3894 Mean F1 = 0.3957 Mean IoU = 0.31904 Training Loss = 0.4801 Validation Loss = 0.5960

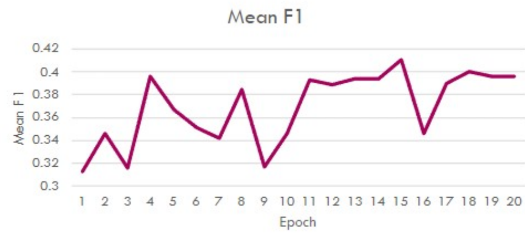


Fig. 14. Mean F1 for the Semantic Segmentation

Fig. 15. Training Loss for the Semantic Segmentation

B. User Study Results

The study is distributed in 5 tasks. As part of the study, The user will be controlling the endoscope camera in the Unity 3D software present on the simulated DaVinci Research Kit. In the first three tasks, the user will be seeing 4 cylinders present on the ”Operating Table”. The job is to center the viewpoint on the 4 cylinders, till they turn a different color. The next task, the user will be seeing an image and a semantically segmented color map of the image joined together. The user will be asked to track a specific color within the image and the researcher present with the user shall monitor if the user has passed or failed the task. In the final task, an actual surgical image will be shown and the region of interest will be marked. The users are required to Follow the gaze as best as they can to center the screen on the region of interest.

The results were not favorable to gaze tracking, however this may have been at the fault of the user study design. Figure 17 through 26 shows the results of the survey the participants took after our study. The most positive results

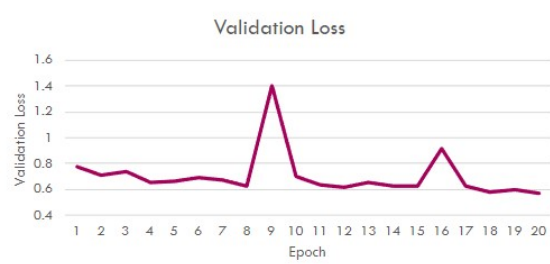


Fig. 16. Validation Loss for the Semantic Segmentation

for the first experiment of putting the central dot over the four cylinders seems to be the key inputs. The following gaze tracking experiments had more favorable results implying that there was a learning curve to the system. If we were to run the experiment again we would give the participants time to get familiar with the system before performing the experiments. The calibration for the users also should have been more controlled. We did not realize that we would have to re-calibrate every user at the beginning of our experiment so we had to improvise how to do this. We would hold the mouse at around the point we wanted the camera to move based on gaze and update the code so that it would move at this point. There were some issues with this as us placing the mouse was not exact and while when I calibrated for the program myself I went through multiple adjustments but only had one value taken for the users. In addition, we determined the user was on the cylinders by when a small dot in the center of the screen was on top of the cylinder. This worked well for keys and mouse; however, how our gaze tracking works is that the camera would move until the gaze is in the central cell, not necessarily the center of the screen. This required our users to look away from the object out of the central cell in order to move the absolute center of the camera to the object. This slows them down and makes the system less intuitive. If we were to do the experiment again, we would require to user to get the cylinder into the central cell for all three methods.

VII. Future Work

In addition to formatting a better user study for the metrics we tried to test in this experiment, finding out the ideal speed of the camera movement with gaze tracking is another test to be done. There were times that it felt like the camera was moving too fast for the users to be comfortable with, especially when unfamiliar with the system. Having the speed be set at the most comfortable speed for our user would increase the comfort level and usability of our system.

Challenges

- Issues faced while importing URDF into the unity due to incompatible version mismatch.
- There were some issues with respect to setting the pre-processing steps for semantic segmentation.
- Tobii Eye Tracker System was only compatible with windows operating system.
- End to end ROS integration has some issues with respect to package import, assembly reference and real time connectivity issues.

We plan to extend the current work to integrate ROS and Unity end-end to achieve better movement of Patient side manipulators in coordination with Endoscope Camera Manipulator with gaze based movement so that we can test the current scope of the work for various surgical scenarios. Also, we plan to use OpenCV based vision algorithms to achieve the region of interest mapping. We plan to integrate both the systems to have a combined system to achieve better coordination between gaze tracking, manipulator movement as well easier visualization for users to improve better viewpoint and reduce operator workload and fatigue levels. Also, we plan to automate the Tobii Eye Tracker 5 within unity environment.

Acknowledgment

We would like to express our gratitude to Prof. Jane Li for facilitating an intellectual dialogue on robotics and artificial intelligence. We are grateful for providing us with the Tobii Eye Tracker 5, which was of utmost importance. Last but not the least, we thank all the Teaching Assistants for helping us with the project setup and through to the execution.

References

[1] M. Minamoto, Y. Suzuki, T. Kanno and K. Kawashima, "Effect of robot operation by a camera with the eye tracking control," 2017 IEEE International Conference on Mechatronics and Automation (ICMA), 2017, pp. 1983-1988, doi: 10.1109/ICMA.2017.8016122.

[2] S. Miura, R. Ohta, Y. Cao, K. Kawamura, Y. Kobayashi and M. G. Fujie, "Using Operator Gaze Tracking to Design Wrist Mechanism for Surgical Robots," in IEEE Transactions on Human-Machine Systems, vol. 51, no. 4, pp. 376-383, Aug. 2021, doi: 10.1109/THMS.2021.3076038.

[3] Roch PJ, Rangnick HM, Brzoska JA, Benner L, Kowalewski KF, Müller PC, Kenngott HG, Müller-Stich BP, Nickel F. Impact of visual-spatial ability on laparoscopic camera navigation training. Surg Endosc. 2018 Mar;32(3):1174-1183. doi: 10.1007/s00464-017-5789-1. Epub 2017 Aug 24. PMID: 28840317.

[4] Huber T, Paschold M, Schneble F, Poplawski A, Huettl F, Watzka F, Lang H, Kneist W. Structured assessment of laparoscopic camera navigation skills: the SALAS score. Surg Endosc. 2018 Dec;32(12):4980-4984. doi: 10.1007/s00464-018-6260-7. Epub 2018 Jun 4. PMID: 29869085.

[5] T. D. Col, A. Mariani, A. Deguet, A. Menciassi, P. Kazanzides and E. De Momi, "SCAN: System for Camera Autonomous Navigation in Robotic-Assisted Surgery," 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2020, pp. 2996-3002, doi: 10.1109/IROS45743.2020.9341548.

[6] A. Mariani et al., "An Experimental Comparison Towards Autonomous Camera Navigation to Optimize Training in Robot Assisted Surgery," in IEEE Robotics and Automation Letters, vol. 5, no. 2, pp. 1461-1467, April 2020, doi: 10.1109/LRA.2020.2965067.

[7] Twinanda, A. P., Shehata, S., Mutter, D., Marescaux, J., De Mathelin, M., Padoy, N. (2016). "EndoNet: a deep architecture for recognition tasks on laparoscopic videos", IEEE transactions on medical imaging, 36(1), 86-97.

[8] Zhou, Z., Rahman Siddiquee, M. M., Tajbakhsh, N., Liang, J. (2018). Unet++: A nested u-net architecture for medical image segmentation. In Deep learning in medical image analysis and multimodal learning for clinical decision support (pp. 3-11). Springer, Cham.

[9] Hong, W. Y., Kao, C. L., Kuo, Y. H., Wang, J. R., Chang, W. L., Shih, C. S. (2020). Cholecseg8k: a semantic segmentation dataset for laparoscopic cholecystectomy based on cholec80. arXiv preprint arXiv:2012.12453.

[10] https://www.kaggle.com/datasets/newslab/cholecseg8k

Most users felt Gaze and key inputs were both equally intuitive to center the gaze
Users felt that gaze required less effort compared to key input
We obtained mixed responses from users regarding the usage of level of intuitiveness when using mouse based interface
Users – felt color based centering based region of interest was easier than color based centering
Well Calibrated system- with specific user calibration was better and easier to use than using single time calibration
Some users felt the position of camera with respect to screen was disturbance based on their height mismatch with respect to gaze tracker position

TABLE I
USER STUDY INFERENCE.

Size	285 mm wide
Housing	Machined aluminium
Shape	Flat front
Sensor	1S5 with custom Tobii NIR sensor (850nm)
Field of View	40*40 Degrees
Supported screen size area	$15''to27''[16:9]or30''[21:9]*$
Head tracking	CPU + Neural Network (CNN) combined / 6DoF**
Image sampling rate and gaze frequency	133Hz non-interlaced gaze at 33Hz
Illuminator	33Hz
Gaze recovery	Continuous recovery
Biometric security	Windows Hello 4.x using NIR + RGB
Software	Tobii Experience
System requirement	Windows 10/11
System recommendations	6th gen Intel Core (i3/i5/i7-6xxx), 64 bit processor. Min 2GHz, 8GB RAM, USB port

TABLE II
TOBII EYE TRACKER 5 HARDWARE SPECIFICATIONS