AR-Assisted Medical Training: Tutorial Generation & Eye Gaze Tracking Analysis

Team Members:
Prateek Bhatnagar and Allan Wang

Mentors:
Ehsan Azimi, Chien-Ming Huang, Ph.D., Peter Kazanzides, Ph.D., and Nassir Navab, Ph.D.

601.446/646 Advanced Computer Integrated Surgery II
Spring 2018

Abstract

Currently, there is a framework for developing augmented reality (AR) training. This framework is based on optical see-through head-mounted displays (OST-HMDs) for the training of caregivers in medical environments. The interface provides visual aids, including images, text, and tracked 3D overlays for various tasks. Our project seeks to create a software tool to facilitate the semi-automated creation of these medical training tutorials. This content generator will use the Microsoft HoloLens's voice recognition and image capture capabilities to record necessary data in a procedure and provide an intuitive interface for anyone to create step-by-step instructions in AR. Furthermore, software API will be created to facilitate 3D and 2D heatmap generation with eye gaze tracking data — collected using Pupil Labs HoloLens add on — and a visualisation will also be created to represent the data in order to aid the optimisation of logistics.

Goal

To create a software tool to facilitate the semi-automated creation of AR-based medical training and allow for gaze tracking data collection during training.

Background

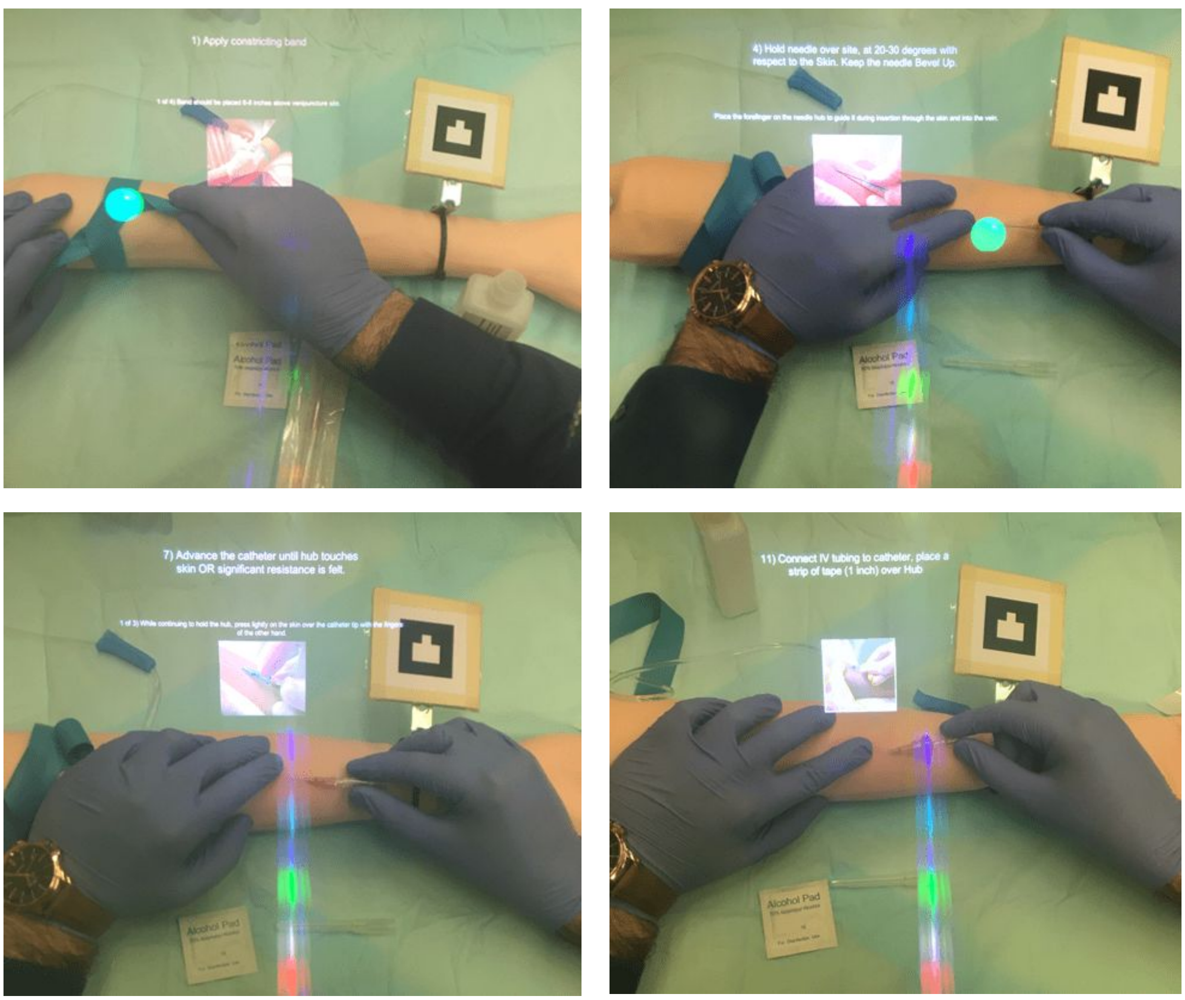


Figure 1. Current appearance and layout of AR training procedure.

HMDs have been employed in medical treatment, education, rehabilitation, and surgery. Graphical layers on an OST-HMD can display relevant data, images, text, or videos and thus facilitate procedures that would otherwise require the practitioner to refer to a side display, turning his or her gaze away from the object of interest. HMDs not only facilitate these processes, but also are relatively unobtrusive as they may be controlled through voice commands.

These devices are best used in controlled training environments due to potential dependencies on internet and appropriate lighting conditions for the optical display. Moreover, the virtual markers used in current training files rely on static visual landmarks that the device uses to determine the location of the markers. If a training file involves virtual markers, it is vital for the user or trainer to ensure that visual landmarks are placed consistently and visible to the Hololens camera.

Technical approach

The HoloLens Development requires Visual Studio with the Windows 10 SDK along with the Vuforia extension to work with Unity. The content generation modules will be developed within Unity. Our first step in development will need to include initial directions for the user in order to navigate the user interface and begin content generation. These menus will be controlled through voice commands, since gestures in the current HoloLens libraries are relatively limited and most trainees will be fully occupied with the task.

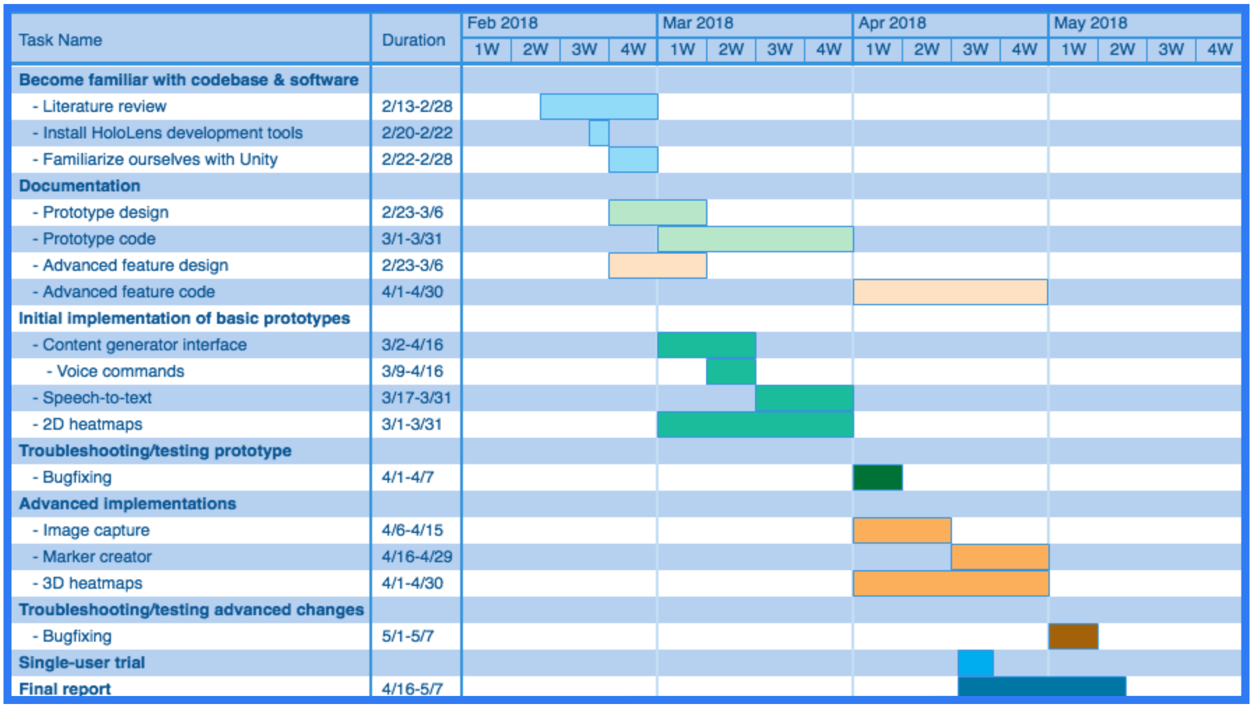
Unity for the HoloLens offers built-in speech recognition modules that can use either online or offline libraries to convert speech to text. While recording speech input, there will be a visual indicator for active recording.

Unity also is able to directly access the camera in the HoloLens device, although it may only be used by one process at a time, and would prevent our program from capturing video and images simultaneously. After image capture, the image will be displayed to the user for approval, retake, or rejection.

The recorded text will be written to a JSON file and images will be saved as they are captured and approved. Recorded audio will also be saved so necessary corrections can be made manually to the text afterwards.

For the heat map generation the API will primarily be written using Python 3 as the software layer layer in the Pupil software stack also uses it. Depending on the software stack structure and existing APIs of the Pupil our layer will be designed around it. The computing for gaze tracking will be offloaded to an separate computer. There many visualisation libraries that can be used in conjunction with Python for 2D heatmap data; however, for 3D data, we may implement a visualisation system that uses the HoloLens itself.

Timeline



Development will begin with the content generator's user interface. The UI will be structured such that data recording modules can be easily added. "Slots" will be created for text, image, and marker data creation. Each module will be individually completed before adding the next. Modular development will ensure that the final version will have at least one of the three distinct functions.

Gaze tracking will be implemented as a parallel process during training *usage*. Development of this module will be completed in parallel with the content generation program.

Milestones

March 18	March 31	April 15	April 28	May 6
User Interface	**Text-to-Speech & 2D Heatmaps** *(Minimum deliverable)*	**Image Capture, Working Demo & Single User Trial** *(Expected deliverable)*	**Marker Creation & 3D Heatmaps** *(Maximum deliverable)*	**Final Report & Demo**
• Able to accept voice commands • Gaze tracker ynchronizing with video feed	• Able to generate text-based tutorials • Gaze-tracking implemented with 2D heatmaps	• Tutorials include both text and images • Have neurosurgeon create training module	• Able to create virtual markers • 3D heatmaps added	• Software ready for live demonstration • Final report and presentation

Deliverables

Minimum	● Working demo of tutorial editor ○ *Speech-to-text* ○ Generation of *2D heatmap* of gaze
Expected	● Working demo of tutorial editor ○ Speech-to-text ○ *Image capture* ● Generation of 2D and 3D heatmap of of gaze ○ Ability to view heatmap data in using a graphical aid - either within the Hololens or some interactive 3D environment
Maximum	● Working demo of tutorial editor ○ Speech-to-text ○ Image capture ○ *Marker creation* ○ *Expertise levels* ● Using 3D and 2D gaze tracking heatmaps to optimize processes ● Testing with ventriculostomy procedure under guidance of medical professional

Dependencies

Dependency	**Requirements**	**Status**
Access to HoloLens & Pupil Labs HoloLens Binocular Add-on	● Submit LCSR form for Robotorium access ● Contact Dr. Huang for access to his lab	Complete
Access to existing codebase (Github)	● Email Ehsan for Git invite	Complete
Installation of toolkits and Unity SDK	● Visual Studio ● Hololens Emulator ● Unity ● Vuforia	Complete
Neurosurgeon to try out a demo	● Get contact info at next meeting with mentors - February 27 ● Email/call neurosurgeon to set up date - March 1 ● Have single-user trial of demo - Mid-April ○ Ventriculostomy	In progress

Management plan

- Weekly meetings with collaborating team
    - Tuesdays after class (3pm)
    - Discuss integration
    - Review compatibility
- Biweekly meetings with mentors
    - Updates on progress
    - Conflict and dependency resolution
- Source control
    - Develop on a branch of existing codebase
    - GitHub repository

Reading list

1. Evaluation of Optical See-Through Head-Mounted Displays in Training for Critical Care and Trauma.
2. Microsoft Mixed Reality: https://developer.microsoft.com/en-us/windows/mixed-reality/
    a. Mixed Reality Tools: https://developer.microsoft.com/en-us/windows/mixed-reality/install_the_tools
    b. Microsoft Mixed Reality Academy: https://developer.microsoft.com/en-us/windows/mixed-reality/academy
3. Pupil Labs Documentation: https://docs.pupil-labs.com/

References

1. Evaluation of Optical See-Through Head-Mounted Displays in Training for Critical Care and Trauma.
2. Kato, H., & Billinghurst, M. (1999). Marker Tracking and HMD Calibration for a Video-Based Augmented Reality Conferencing System. In Proceedings of the 2Nd IEEE and ACM International Workshop on Augmented Reality (p. 85--). Washington, DC, USA: IEEE Computer Society. Retrieved from http://dl.acm.org/citation.cfm?id=857202.858134
3. Birt, J., Cowling, M., & Moore, E. (2015). Augmenting distance education skills development in paramedic science through mixed media visualisation.
4. Armstrong, D. G., Rankin, T. M., Giovinco, N. A., Mills, J. L., & Matsuoka, Y. (2014). A heads-up display for diabetic limb salvage surgery: a view through the google looking glass. Journal of Diabetes Science and Technology, 8(5), 951–6. https://doi.org/10.1177/1932296814535561
5. Tai, B. L., Rooney, D., Stephenson, F., Liao, P.-S., Sagher, O., Shih, A. J., & Savastano, L. E. (2015). Development of a 3D-printed external ventricular drain placement simulator: technical note. Journal of Neurosurgery, 123(4), 1070–6. https://doi.org/10.3171/2014.12.JNS141867
6. Atkins, M. S., Tien, G., Khan, R. S. A., Meneghetti, A., & Zheng, B. (2013). What do surgeons see: capturing and synchronizing eye gaze for surgery applications. Surgical Innovation, 20(3), 241–8. https://doi.org/10.1177/1553350612449075
7. Kersten-Oertel, M., Jannin, P., & Collins, D. L. (2012). DVV: a taxonomy for mixed reality visualization in image guided surgery. IEEE Transactions on Visualization and Computer Graphics, 18(2), 332–52. https://doi.org/10.1109/TVCG.2011.50
8. Eck, U., Stefan, P., Laga, H., Sandor, C., Fallavollita, P., & Navab, N. (2016). Exploring Visuo-Haptic Augmented Reality User Interfaces for Stereo-Tactic Neurosurgery Planning. In G. Zheng, H. Liao, P. Jannin, P. Cattin, & S.-L. Lee (Eds.), Medical Imaging and Augmented Reality (pp. 208–220). Cham: Springer International Publishing.