

Using AR/VR to Train Perioperative Teams to Identify and Communicate Safety Hazards in  
Their Environment

Jason Phadnis (ID: 5138141)

EE 4982V

Advisor: Prof. Daniel Keefe

External Collaborator: Dr. Cullen Jackson

Spring 2021

## Table of Contents

<b>Abstract</b>	<b>2</b>
<b>List of Figures</b>	<b>2</b>
<b>1. Introduction</b>	<b>3</b>
<b>2. Background</b>	<b>5</b>
<b>3. Iterative Design Methodology and Lessons Learned</b>	<b>6</b>
<b>3.1: Iteration 1 - The Virtual Room</b>	<b>7</b>
<b>3.2: Iteration 2 - Hazard Setup App</b>	<b>9</b>
<b>3.3: Iteration 3 - Hazard Zones</b>	<b>11</b>
<b>3.2: Iteration 4 - Integrating Hazard Type</b>	<b>13</b>
<b>4. Results</b>	<b>16</b>
<b>5. Conclusion</b>	<b>20</b>
<b>6. Future Work</b>	<b>20</b>
<b>References</b>	<b>22</b>

## **Abstract**

The arrangement of equipment and other subtle details within an operating room can create unnecessary hazards for the doctors, nurses, and patients during an operation. Here we sought to discover and understand to what degree could VR facilitate learning, both individually and as a group. To this end I worked on a VR application that can allow one to explore a virtual operating room, see potential hazards, and mark hazards for both themselves and others. Once completed, the application can be used to train doctors and nurses to recognize hazards within their own operating room and take steps to avoid them, both in preparing the room and during an operation. We scanned a real operating room and made it interactable in an Oculus Quest VR headset. We focused on developing the system iteratively, judging it feature by feature and gaining a better sense of what we wanted along the way. Through this process we learned that we wanted the user to be able to define areas as hazardous which could enclose the specific offending item(s). Furthermore, these markings must portray a certain amount of information at a glance, and ideally more when the user is curious about any marking in particular. Developing these features also informs our vision of how the users will use the application both individually and how to use it as a catalyst for group discussions.

## **List of Figures**

1. Figure 1. A room scan obtained from Canvas and placed into Unity. (pg 8)
2. Figure 2. Hazard objects with description text and a laser pointer coming from the right controller. (pg 10)

3. Figure 3. Grabbing a hazard. (pg 10)
4. Figure 4: An early version of the hazard type and severity selection table. The sphere is currently colored yellow for the lowest severity level. (pg 13)
5. Figure 5: The iteration where the hazard zone icon representing the hazard type and severity level are tiled across the sphere's surface. These spheres suffered from being hard to distinguish from one another when bunched up. (pg 14)
6. Figure 6: The latest iteration for the hazard zone spheres. The placement of the single icon for each sphere makes them easier to distinguish from one another. (pg 15)
7. Figure 7. The demo operating room with some other 3D models placed in as hazards. (pg 16)
8. Figure 8. The user is creating a hazard sphere in the area of the cart and the patient. There is also another hazard to the left in the back of the room with the red symbol. (pg 17)
9. Figure 9. An example of a hazard zone created by another user that has an explanation as to why they marked the object as a hazard. (pg 18)
10. Figure 10. An example of a hazard from another user that doesn't have a description. The current user didn't think to mark this, so they may wonder why someone else did. This can become a discussion topic. (pg 19)

## **1. Introduction**

In medical operating rooms, simple placement of items around the room can create unnecessary hazards. For instance, a defibrillator can be rolled in but then blocked off by an IV stand. During an operation, if this defibrillator is needed, the IV stand needs to be moved while



still connected to the patient. At best, this wastes a few seconds trying to get the defibrillator, and at worst the IV is disconnected from the patient which can cause other problems during an operation. In addition to access being blocked by equipment, hazards include difficulty accessing the patient based on positioning and low light levels making it hard to read labels [1].

While safety experts can come in and identify such safety hazards, this doesn't solve the root problem. The doctors, nurses, etc who work in the operating room should be trained to identify safety hazards themselves and take steps to avoid them. To this end, I worked on a VR system that would allow surgical staff to tour their operating room with different hazards in place. In this training application, they can see information as to what is hazardous and why.

First we (i.e. Prof. Keefe, Dr. Jackson, and I) needed to determine what the requirements and features for the application should be. This meant iteratively building up a prototype to test different features and different variations on individual features. Through this we gained a better understanding of what was feasible and refined the requirements to fit not only what we could do but what would be the most effective. This started with getting a scan of an operating room to serve as our 3D VR environment. Then I worked on getting text to display floating in the air to be used as annotations for hazards. Pivoting I tried developing a system to easily place hazard objects in the room which have annotations and could be manipulated within VR. Pivoting again, I implemented a system to create zones to represent hazards. These zones have different colors and symbols associated with them to provide some description of the hazard. I also have the ability to save these zones and load them back in.

## 2. Background

Related AR work includes research into AR as an educational tool. AR promotes learning; however technical problems and reduced usability hampers AR's success in this regard [2]. Another paper, comparing traditional 2D interfaces for teaching and the MaxWhere 3D VR platform, found that teacher's require more familiarity with technology for the VR/AR platform. In return though, students and teachers are able to complete tasks in more ways and require 37-64% less user operations and upwards of 72% less machine operations [5].

Based on a study in 2019, in the medical field AR technology is being used more for treatment than training. Displays are already widely used in operating rooms; this means integrating AR into the workflow is relatively easy as it is simply a different display. On the training side, surgery simulations were a common use for AR. A hindrance to AR's effectiveness in training may be the bulkiness of the headsets because there were relatively few AR cases involving collaboration [3]. Another study on VR and AR in medical education found that these technologies are capable of reducing the effects of unskilled operation and decreasing costs for medical training. Main problems include technical difficulties and the integration of this technology with the nursing and health care information system [4].

AR has also been used by NASA to train astronauts in an extreme underwater environment. In the study, the astronauts had to complete a complex series of tasks that took place across four different locations. In the end the astronauts reported that the AR proved helpful in completing their tasks in which they were given notes similar to what we hope to

achieve. The astronauts also said the AR helped with finding where to go, but paper instructions are more easily navigable due to the ability to jump to a certain section [6].

Looking at all of this previous work, AR and VR has been used in educational training, assisted in medical procedures, and used for some medical training. However the medical training has been more for procedures such as surgeries. There is no training experience focused on searching for and removing the small and common hazards that can be found in operating rooms. Therefore, I sought to fill that void and create a method of increasing the safety of the medical workspace by building my application. Furthermore, while this particular application focuses on the medical field, the topic of hazards can be extended to other fields. There are many jobs, including medical staff and astronauts, that have safety hazards that workers need to be wary of. More than that though, the method of learning itself is potentially very useful, especially with the current online setting.

### **3. Iterative Design Methodology and Lessons Learned**

At the beginning of this project, the idea was simply creating an AR/VR application to help people learn about safety in the space where it matters. The main question was then to what degree does such an application improve learning, facilitate group discussion and learning, and how to run such an application given how headsets are necessary and meeting in person is harder with the Covid-19 pandemic. At the time we had only a high level set of requirements and no clear plan as to how to implement them. However, throughout the process of building this application we gained a great deal of knowledge to help us move forward. We started by working out what requirements we could before diving into making a prototype application. First, since

the Covid-19 pandemic makes it harder to have in-person group meetings, AR would be harder to test. VR meanwhile presented a possibility of remote training; therefore I rented an Oculus Quest VR headset to develop the application. Furthermore, we wanted the user to walk around a 3D operating room and identify objects as hazards. This involved somehow selecting objects that would then be highlighted and have a text annotation floating next to it if that object was a hazard. The application should also be able to allow users to learn as a group and create a better shared mental model of safety hazards and the ways to avoid them.

### **3.1: Iteration 1 - The Virtual Room**

With some basic decisions made I started working on the prototype application. I made changes to the prototype iteratively and got feedback from Prof. Keefe and Dr. Jackson via emails and/or meetings on a weekly basis. Thus I was able to benefit from both Prof. Keefe's technical expertise and Dr. Jackson's knowledge of the medical field and clinical team coordination. Since we decided on VR, the first major goal was to create a virtual operating room that a user could traverse. We began by using the Canvas software by Occipital to scan a room and generate a 3D model of it. I then used Unity Technologies' Unity Engine to make this room interactable via the Quest. Once the room was successfully loaded onto the Quest, I worked on basic player movement and representation of the player's controllers.

However, while in the process of creating rooms to be used for training, we found there was a problem with the scan quality. Figure 1 below shows an example of one of the scans we obtained. We see that some parts are blurry or otherwise don't look as realistic. The blurring appears to be from moving the camera past the room a little too quickly. Then there are also

holes in the room scan. As it turns out, it is hard to get the entire room in one scan without leaving parts out and thus creating holes.



Figure 1. A room scan obtained from Canvas and placed into Unity.

To help combat these problems, we tried getting multiple scans of the room to get the parts behind the machines and objects in the room. However, even after I stitched together the latest 3 scans using the Blender 3D modeling software, there were still some holes. Furthermore we found it difficult to move the camera at a speed when scanning that prevents blurring. Overall the room scan can be rather distracting. Therefore, we would either need to find a different method of getting a 3D model of a operating room, or we would have to devise a way to scan the entire room slowly enough to prevent blurring and thoroughly enough to prevent holes.

### 3.2: Iteration 2 - Hazard Setup App

While we were able to patch a few scans together to cover some of the holes, the room quality was still distracting. Therefore we decided to explore an alternative option for the VR application; using it to inform an AR application. The idea was to have a user put on the Quest and use that to move around the virtual operating room with the hazards in there. Then the user could use the Quest's controllers to move the hazards around the room, resize them, and rotate them. Then we could save some hazard data, such as the position and rotation, to a JSON file which the AR application would use to place the appropriate hazards in the same manner.

I thus created a hazard prefab object in Unity that is easily replicable. It is also rather easy to give each individual hazard a unique model and text annotation floating next to it while still allowing all the hazards to share certain functionality. There is a laser pointer on the right controller, as depicted in the below Figure 2, to help show what the user is pointing at and would therefore select at any given time. While this laser is pointed at a hazard, the user can hit a button to toggle the visibility of the hazard description that appears in the air next to the hazard. One can grab hazard objects, pulling them to the player's hand as seen in Figure 3 below. Then the user could have it offset from the hand to place it in distant and/or hard to reach places. Furthermore the player can rotate the held hazard and change its size to better fit the room setup they envisioned. Finally I created the system to save the position, size, and scale of these hazards in a JSON format.



Figure 2. Hazard objects with description text and a laser pointer coming from the right controller.



Figure 3. Grabbing a hazard.

As we worked with these interactable hazard objects, another concern started to come to light. We learned that while it's still good to have annotations accompanying the objects in the room, we also don't want them appearing all at the same time. If there were many hazards clumped together each with their own description, then it could prove to be too difficult to read some of the text. We realized that really a user would only want to see the information for one or two hazards at any given time. Therefore in the next iteration we selectively hide and unhide the text to only show what is relevant at the time.

### **3.3: Iteration 3 - Hazard Zones**

Reviewing this hazard setup application, we decided that it would be better to develop the application following a new idea. Instead of having the user place hazards or try to select hazard objects that would then reveal themselves like our original idea, we decided to focus on marking areas. More specifically a user could create a spherical zone which could encompass what the user believed to be a hazard. Another requirement was to have the application able to save these zones and redisplay them in future sessions of the application. Thus one user could see what other users marked as hazards and start wondering why. The idea was that this would then lead to discussion among the users and help train an entire group.

I thus started working on the system to create hazard zones instead of individual hazard objects. We realized that the user must be able to select what they believe to be hazards quickly and intuitively. Thus I gave the user the ability to, with the click of a button, create a sphere at the point that the user's laser pointer is hitting. The user resizes this sphere simply by moving the laser pointer away from the point the user clicked on. This makes it easy and intuitive for users



to create a zone that can encompass objects of various sizes and as many objects as they so choose. I then made it possible to reposition and resize the sphere to correct and/or better place their marker.

Throughout the development of this system, we imagined how the clinical staff would use the application. In particular, we sought a way to invite discussion and collaboration between users with our application, and we figured out that being able to see what the user's colleagues think are hazards is an important first step. Merely seeing something marked that a user missed can invoke questions followed by conversations about hazards. To this end I made the system able to save data on the hazard zones in the form of JSON which is stored on the Quest. When the application is opened up again, all the zones created in previous sessions are recreated as they were. Furthermore, I created a text display to appear in front of the user's view like a heads up display (HUD) to show certain information. For instance, when a user points at a zone created in a previous session, the text that appears shows which user created it and the date they did so. This is how we show only the text that the user wants at any given time. A message will appear when the user successfully saves the hazard zones they created or inform them that there is nothing to be saved. I also gave the ability to delete all saved hazards, which has an accompanying message.

Originally, the hazard zones were just semi-transparent red spheres. These are able to show that something is hazardous, but it doesn't tell as much information as it could. We decided to use color to portray the severity level of the hazard. Thus I ended with using yellow for low level, light orange for medium, dark orange for major, and red for critical hazards. However there was yet more information we wished to display with these zones. Next I created a table of

buttons. Each column consisted of a symbol indicating a certain type of hazard such as fire or loud sound. Each row meanwhile indicated different severity levels using the same colors I used previously. Figure 4 below shows an earlier version of this feature.



Figure 4: An early version of the hazard type and severity selection table. The sphere is currently colored yellow for the lowest severity level.

### 3.2: Iteration 4 - Integrating Hazard Type

The way we displayed the selected hazard type went through a few iterations. At first I had the selected hazard symbol, with the correct severity color, tiled across the sphere as seen in Figure 5 below. However we decided that when multiple spheres were next to each other or intersecting, it was hard to tell which zone was which. Thus I changed the type representation to a single central hazard type icon that would rotate towards the player while the sphere had x's to

more easily define the sphere's surface. It was much easier to tell different hazard zones apart with this method. Unfortunately it also had the problem of the central icon often being partially within, and thus obscured by, nearby geometry. The x's on the sphere surface were also hard to see in dimmer light. Finally I settled on having the type icon display in front of the sphere, similar to the hazard type selection table, and have the sphere always rotate to face the user; Figure 6 depicts this below. This version helps prevent the symbol from being obstructed. Thus we obtained a way to convey some information regarding the hazard zone in a natural way. The hazard type and severity is also among the hazard zone data, so the zones are still recreated correctly in future sessions.



Figure 5: The iteration where the hazard zone icon representing the hazard type and severity level are tiled across the sphere's surface. These spheres suffered from being hard to distinguish from one another when bunched up.



Figure 6: The latest iteration for the hazard zone spheres. The placement of the single icon for each sphere makes them easier to distinguish from one another.

Another feature is the ability to load in and populate the room with models to represent realistic hazard objects that weren't in the original room scan. Thus it is easier to create a variety of hazardous room setups without needing to populate a real operating room and scan it. I then created the ability to add information to both standalone models and areas in the room scan representing specific items. When the user points at this such an object, that information is displayed on the user's HUD. A couple of minor features I added include the ability to delete hazard zones that the current user created to correct their own mistakes. They can also hide and unhide hazards created by other users which allows them to mark their own hazards without influence. When they're done they can reveal the other hazards and see what other people thought about.

## 4. Results

Here I present a demonstration of the application as we might expect a real user to interact with it. Figure 7 below is an example of how our operating room scan could be populated with virtual models of hazards and then annotated by users. We also see the user getting some information about the code cart that they're pointing at. This is an example of how certain objects in the room can be given annotations generated by the system as opposed to a user. This can help clarify what the user is looking at and provide some information about the item or how to handle it. In this case, we see that the code cart should never be obstructed. This is because it contains emergency equipment such as the defibrillator.



Figure 7. The demo operating room with some other 3D models placed in as hazards.



In Figure 8 below, the user is marking what they believe to be hazards. Currently they're marking a cart sitting right next to the patient as a hazard because this is where the surgeon would be standing. Thus this cart would need to be moved to allow them to assess the patient. They've also marked a cart with some equipment on it shoved into a corner of the room. You can see its red icon peeking out on the left of the patient. This is bad because it's hard to access that equipment with the IV stand blocking the only path to it. Even worse, that cart is blocking a door which is never acceptable.



Figure 8. The user is creating a hazard sphere in the area of the cart and the patient. There is also another hazard to the left in the back of the room with the red symbol.

Figure 9 below depicts the room after the user has placed all of their hazard zones and they've revealed the zones from previous users. Currently the user is pointing at one of the

hazard zones created by a previous user. The current user sees some basic information such as who created it and what type of hazard it represents. However there is also an example of a description left by the person who created the zone explaining why they marked the object as a hazard. In this case, they think the anesthesia machine that they marked could be distracting due to the noises it can make. Closed loop communication is important among the perioperative team to ensure a safe, successful operation.

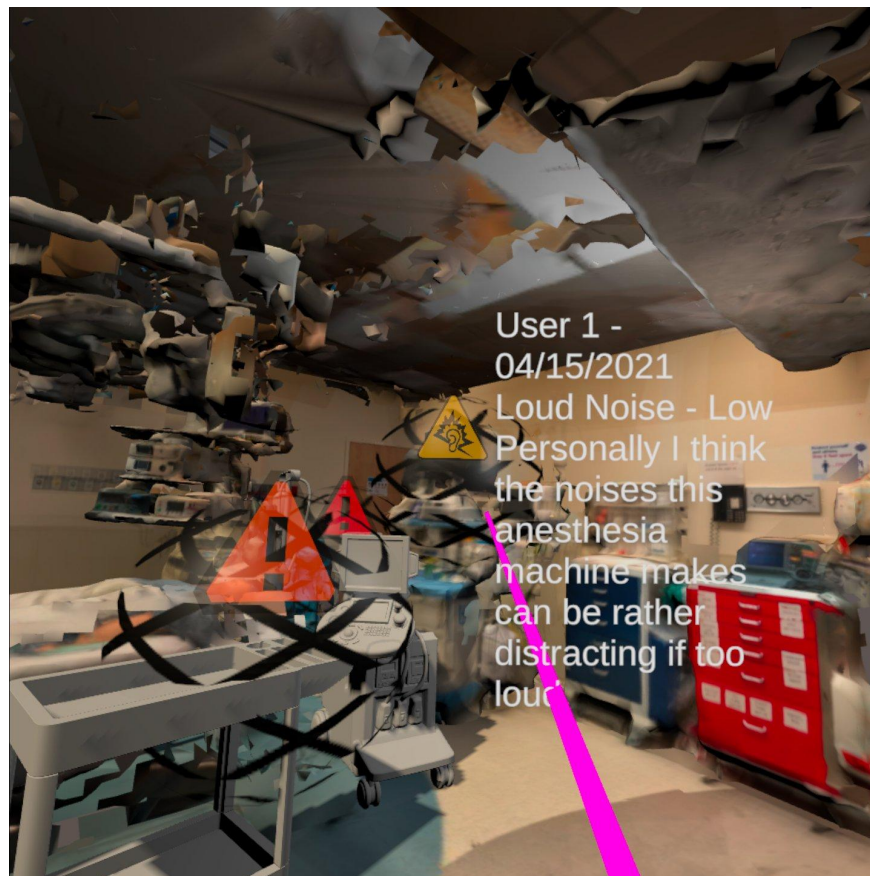


Figure 9. An example of a hazard zone created by another user that has an explanation as to why they marked the object as a hazard.

Figure 10 also shows zones from other users that neither mark things that the current user marked nor have explanations. This can generate questions such as “why did X think this was a

hazard” or “Why did Y rate this hazard as being so severe.” Such questions can in turn spark discussion among the clinical staff using the application. They can learn from each other and gain a consensus of what are problems. During this conversation, solutions to many of the hazards they identified are also likely to come up; therefore the meeting serves to help everyone involved learn about how to make their own work environments even safer.

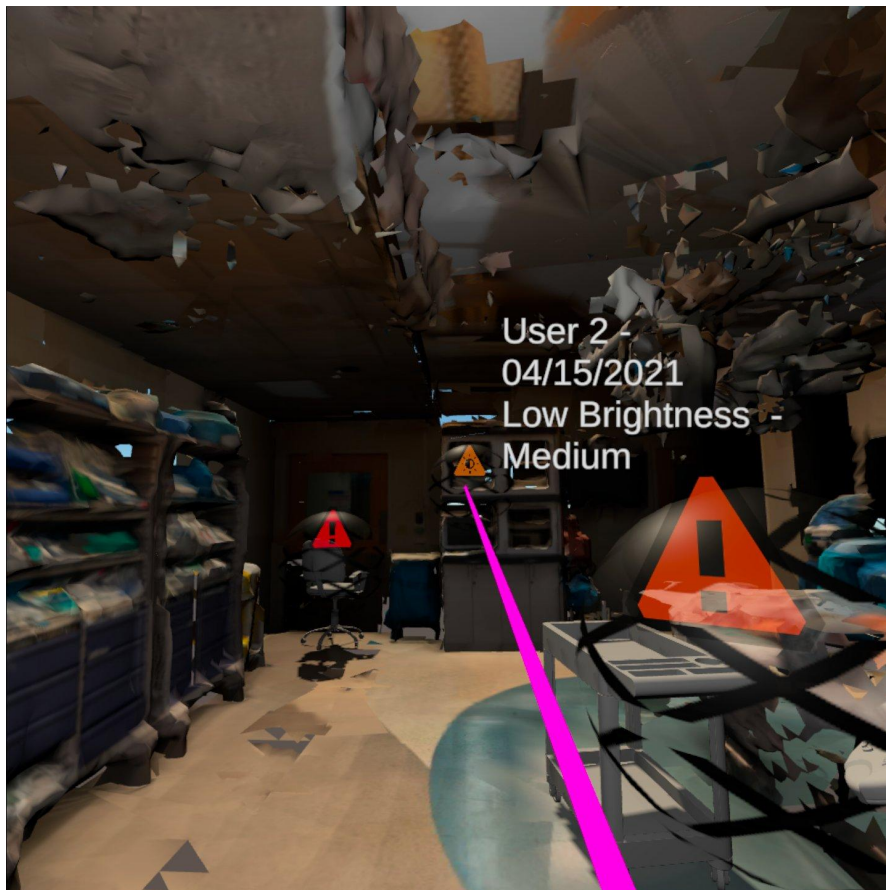


Figure 10. An example of a hazard from another user that doesn't have a description. The current user didn't think to mark this, so they may wonder why someone else did. This can become a discussion topic.



## **5. Conclusion**

We created a prototype VR application for two purposes: training perioperative teams to identify hazards in their work environment and to learn to what degree VR can affect one's retention and group collaboration. Through the iterative development of this prototype we've learned about the problems and potential that our application has from the lack of quality on the room scan to the usefulness of intuitively defining areas as hazardous. We now have an example room that portrays what this application can be with a little more work. The ability to set up a room in any number of combinations and the potential of sparking discussion between users about their work environment are both enticing. The challenges we encountered keep us informed as we continue to work towards the final application.

## **6. Future Work**

A major feature we wish for the application to have is the ability for the user to provide a description of the hazard zones they create. Currently we can only provide such descriptions via Unity and not through the Quest. This feature would allow for more learning on the spot, and users could learn a bit about their colleagues' thought process while exploring the virtual room. Furthermore, these can invite questions and deeper discussions with the people who made the zones and annotations. Because pulling up a keyboard and typing out such descriptions in a Quest would be rather tedious, we would instead look into other methods of creating these annotations. One such idea was to have the user able to speak their description and have the application type that out; this would be both fast and intuitive.

Another major milestone would be having a user study for our application. This would allow us to gain a tremendous amount of feedback as multiple clinical staff use the app. We will work with unit leadership and frontline staff to gain participants for a short training session without disrupting the clinical workflow. The Oculus Quest will be wireless and run the scene on its own hardware. The participants will start by looking around the virtual room with a few hazard zones already revealed and with annotations. This acts as an introduction and gives the participants a sense of what they will be looking for in future sessions. Everyone will have another 5 minute session afterwards in another room prepopulated with hazards. They will try to find as many hazards as they can and mark them. After these 5 minutes the hazards marked by previous users will be revealed. This is to spark conversation between the participants and get them thinking more critically about their environment. After these sessions, everyone will meet up either in person or virtually to discuss their findings. This will also allow the participants to ask each other why they marked certain items as hazards. Through this the users can learn from the different perspectives of their coworkers.

## References

1. C. D. Jackson, "MAstering Cognitive skills using Collaborative Augmented Reality training for PERioperative Teams (MAgiC CARPET)," working paper, Boston, MA, 2020.
2. Akçayır, Murat, and Gökçe Akçayır. "Advantages and Challenges Associated with Augmented Reality for Education: A Systematic Review of the Literature." *Educational Research Review*, Elsevier, 5 Nov. 2016, [www.sciencedirect.com/science/article/pii/S1747938X16300616?casa\\_token=-e14FK7kNiUAAAAA%3AwtuRJcJj0w8yjtYsD\\_mRj2GieVOzHsVgVYKAh5iRHR2Qyrdul3TN\\_3V6wpGp3Ym54LcGOxJqJA](http://www.sciencedirect.com/science/article/pii/S1747938X16300616?casa_token=-e14FK7kNiUAAAAA%3AwtuRJcJj0w8yjtYsD_mRj2GieVOzHsVgVYKAh5iRHR2Qyrdul3TN_3V6wpGp3Ym54LcGOxJqJA).
3. Eckert, Martin, et al. "Augmented Reality in Medicine: Systematic and Bibliographic Review." *JMIR MHealth and UHealth*, JMIR Publications Inc., Toronto, Canada, 2019, [mhealth.jmir.org/2019/4/e10967/](http://mhealth.jmir.org/2019/4/e10967/).
4. Hsieh MC, Lee JJ (2018) Preliminary Study of VR and AR ĐĐÛŝĀtiŽnĚ in Medical and Healthcare ĚŽcĀtiŽn Ĵ Nurs Health Stud Vol. 3:No.1:1.
5. I. Horváth, "Evolution of teaching roles and tasks in VR / AR-based education," 2018 9th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), Budapest, Hungary, 2018, pp. 000355-000360, doi: 10.1109/CogInfoCom.2018.8639907.
6. Nuernberger, Benjamin, et al. "Under Water to Outer Space: Augmented Reality for Astronauts and Beyond." *Computing Edge*, June 2020, [ieeecs-media.computer.org/media/marketing/cedge\\_digital/ce-jun20-final.pdf](http://ieeecs-media.computer.org/media/marketing/cedge_digital/ce-jun20-final.pdf).