Dartmouth Computer Science Technical Report TR2020-897

VR-Notes: A Perspective-Based, Multimedia Annotation System in Virtual Reality

Justin Luo
Dartmouth College, justin.luo.20@dartmouth.edu

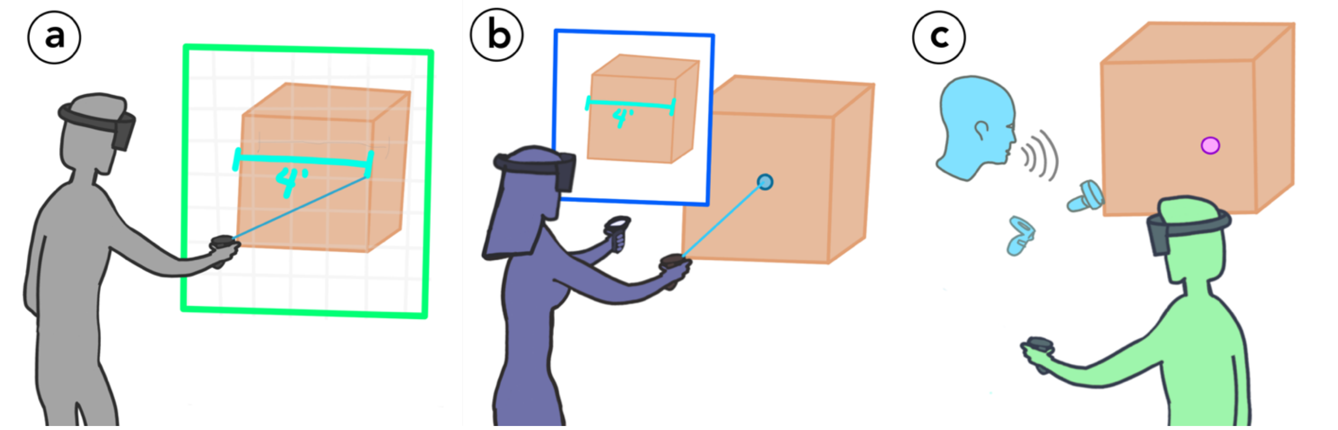

Figure 1. (a) Drawing a doodle annotation in virtual reality on a canvas plane; (b) another user viewing the doodle annotation at a later time, with a perspective background captured from original annotator; (c) playback of audio annotation in virtual reality through an animated avatar based on recorded movements from annotator

ABSTRACT
Virtual reality (VR) has begun to emerge as a new technology in the commercial and research space, and many people have begun to utilize VR technologies in their workflows. To improve user productivity in these scenarios, annotation systems in VR allow users to capture insights and observations while in VR sessions. In the digital, 3D world of VR, we can design annotation systems to take advantage of these capabilities to provide a richer annotation viewing experience. I propose VR-Notes, a design for a new annotation system in VR that focuses on capturing the annotator's perspective for both "doodle" annotations and audio annotations, as well as various features that improve the viewing experience of these annotations at a later time. Early results from my experiment showed that the VR-Notes doodle method required 53%, 44%, 51% less movement and 42%, 41%, 45% less rotation (head, left controller, and right controller respectively) when compared to a popular 3D freehand drawing method. Additionally, users preferred and scored the VR Notes doodle method higher when compared to the freehand drawing method.

Keywords
virtual reality, annotation system, perspective

INTRODUCTION
During the last decade, we have begun to see the rise of VR in the entertainment industry with the advent of affordable and powerful VR headsets and controllers [14]. These VR technologies allow for a level of immersion in video gaming that has not been experienced before, from first person shooters to collaborative social board games. While the introduction to VR for the public may be primarily through gaming and entertainment, many believe that VR will continue to revolutionize how we interact with the digital world in the workplace. We can already see prototypes of how VR technologies can be used in medicine [23, 29], engineering/construction [17, 24], and education [4, 20]. To be more productive in VR, these users will need the ability to annotate the world around them—making observations and taking notes on points of interest in the virtual environment. By doing so, they will be able to record important insights and observations during these VR sessions, especially for other people to view.

Many people are accustomed to annotating or viewing annotations on a paper document or 2D screen like a desktop word processor application. However, annotating in VR is a relatively new experience for most people—many are still learning how to effectively read and write in a 3D space. Studies have demonstrated the challenges that people experience when attempting to draw, write, and read in 3D space [6, 9]. Specifically, the challenges of space and depth perception, as well as the lack of a physical drawing surface, pose significant difficulties to users. As a result, there are opportunities to innovate by implementing an easy-to-use, annotation system that takes full advantage of the immersive 360º environment and capabilities of the digital environment. One opportunity is to explore how we might utilize 2D canvases in a 3D environment to help users draw. Another opportunity is to capture information about the annotator's perspective, by tracking user perspective, location, and movement during the annotation process. Perspective information can then be displayed in an intuitive way with the annotation at later viewing times. By doing so, this could greatly benefit communication and collaborative VR experiences.

Existing research has explored how 2D annotations and metaobjects like circles, arrows, and text might be presented to a user in a 3D environment for use in virtual and augmented reality [15]. Additionally, frameworks have been developed to address the technical challenge of accurately anchoring and rotating annotations in the 3D environment for optimal viewing [25]. While many of these studies focus on transferring aspects of a traditional 2D annotation system to the 3D space, recent studies have begun to explore annotation systems that instead capitalize on using digital 3D space to create richer interactions with annotations in VR [19, 28]. This paper explores a design for an annotation system in that same vein of thought.

In this paper, I present VR-Notes, a design for a multimedia annotation system in VR that captures information about the annotator's perspective, later displaying this information alongside the annotation to provide a more meaningful and clearer annotation viewing experience. VR-Notes highlights two major types of annotations; 1) doodle annotations – hand drawn notes, and 2) audio annotations – voice recording notes. In doodle annotations, I explore the concept of utilizing a 2D plane as the drawing canvas in the 3D environment (Figure 1a), as well as a method of capturing a screenshot of the user's point of view and displaying that as the background for the annotation (Figure 1b). For audio annotations, I illustrate a method of tracking the positions and orientations of user anchor points, and later animating an avatar during audio playback to replicate the annotator's movements during the recording (Figure 1c). This annotation system focuses on both creating and viewing annotations in VR, with a focus on preserving context and providing a consistent viewing experience for users at a later time. Additionally, this annotation system was designed to reduce the amount of unnecessary movement and rotation during a user's annotating process. Thus, this design results in a more comfortable annotating experience, especially during extended use.

While exploring different options for an annotation system, I compared the performance of VR-Notes doodle feature with a popular 3D drawing system found in many popular VR applications as the control method. Because of COVID-19 and social distancing, I was only able to conduct a minimal amount of experiments with the people who lived with me. However, the early results indicate that the VR-Notes doodle method reduced head, left controller, and right controller movement by 53%, 44%, 51% and rotation by 42%, 41%, 45% respectively when compared to the control method. The VR-Notes doodle method also reduced time spent while creating the drawing by 13% compared to the control method. Additionally, early results indicated higher feedback ratings and a preference for the VR-Notes doodle method over the popular freehand drawing method.

In summary, my contributions are: (1) a design for an multimedia annotation system that features both doodle and audio annotations, (2) designs for additional features in the annotation system that improve quality of the experience for annotators and viewers, (3) a user study with early results demonstrating the effectiveness of VR-Notes, and (4) two demo applications that illustrate potential use cases for VR-Notes in education and academic research.

RELATED WORK

My work builds upon previous research and explorations into annotations, with a focus on annotation systems in a 360º digital environment, often in VR or augmented reality (AR).

Annotation Systems in 2D Digital Documents

Early research in annotation systems in technology has primarily focused around systems that can be implemented in digital documents that exist primarily on a 2D screen. One study by Cadiz et al. analyzed the way that users used the annotation system in Microsoft Office 2000 [12]. The researchers identified several different factors that influenced usage of the annotation system, including how responsive their peers were to the annotations, how annotations were correctly anchored, and the public nature of the annotations. Additionally, researchers have explored the use of annotations in digital documents and web-based applications in collaborative environments, including designing for notification systems [7] and robust anchoring systems for annotations [10].

Frameworks for 3D Annotation Systems

Early work by Feiner et al. explored basic interactions of a knowledge-based annotation system in the "virtual-world" [15]. Through a see-through, head-mounted display, the researchers were able to overlay graphics onto the user's point of view, and they designed a system to intelligently place textual annotations and virtual metaobjects (arrows and text) in the user's world. Additionally, the system was able to respond to certain user interactions like object selection. Although this work does not explore the actual process of making annotations in a virtual environment, it contributed to laying the foundation for augmented reality experiences and annotation systems in this area.

Kadobayashi et al. explored annotations in virtual 3D spaces by describing key features of an annotation tool to enable meaningful collaboration [22]. These features included the use of an "interactor," where users have the ability to add annotations and control the view of annotations, and a "history viewer," which gives users the ability to view an annotation's history of modifications over time in a 3D concept map. Pick and Kuhlen also outline a framework for VR annotation systems using adaptable building blocks and discuss methods of integrating new data types in various annotation scenarios [26].

Explorations into Annotation Systems and Drawing Methods in Virtual 360º Environments

As VR and AR headsets and technologies are becoming more widespread available to the public for commercial and research use, researchers have explored their own designs for annotation systems while in these experiences. One study by Jung and Gross detailed a tool called the "Space Pen" allowed users to create post-it style annotations on 3D models, as well as freehand drawing on the models and environment directly [21]. Clergeaud and Guitton designed a system where a document can be retrieved and edited in both VR and real environments, syncing seamlessly as the user changes platform [13]. In another study by Matos et al., researchers explored various types of annotation metaobjects (markers, circles, text) in a 360º environment and gauged how useful and effective these metaobjects were in giving instructions to user participants [24].

Consumer VR applications like TiltBrush [3] and Quill [1] explore 3D drawing and art, which allows users to create 3D strokes suspended in the air around you. Although these applications are not originally created for annotation purposes, the freehand drawing in 3D can be used as a technique for drawing annotations. Yet, space and depth perception has proved to be a challenge for users in immersive 3D drawing [5], as well as a lack of a physical drawing surface [9], and several techniques have been created to help assist users during the 3D drawing process [7, 8, 16]. Because of the popularity and success of these applications in the VR community, this type of drawing system should be considered when designing an annotation system in VR.

Focus on Collaboration in Virtual 360º Environments

While discussing annotation systems, one must consider how these annotation systems will be used collaboratively among multiple individuals, especially in a platform like VR that still has much to explore in terms of social interactions. Spacetime, a collaborative editing tool designed by Xia et al., presents a detailed system workflow where users can edit objects seamlessly at the same time in VR [30]. The tool utilizes avatar characters to share perspectives during the collaborative experience.

Gauglitz et al. has also explored collaboration where one user is in an augmented reality environment, and another user is using a typical mobile or computer interface [18]. In the system, one user completes a task in the physical world while wearing an augmented reality headset. Video is captured and streamed through the headset to the remote user, who can annotate the physical environment that the local user is in. Utilizing this system, the researchers hoped to enable a more effective way to communicate with remote users performing physical tasks.

Multimedia Annotation Systems in Virtual 360º Environments

As VR headsets include more features like motion tracking, microphone input, and even external camera sensors, there are opportunities to expand beyond the typical text annotation towards multimedia content in annotations in VR. For example, Sutherland created an audio annotation system in VR called Glossy, which uses the headset's microphone to record voice messages, which can then be played back later when the user interacts with an annotation marker [27]. In addition, Sutherland describes future possibilities of utilizing position information to describe the annotator's path while leaving the voice message, as well as the functionality to create threads of voice messages by linking them together.

Guerreio et al. describes a multimedia set of annotation features in their design of an annotation system, with an emphasis of how these annotations might work in a collaborative environment [19]. They look into the idea of a "discussion tree," which links text, audio, and video annotations in a digital tree-like structure. Tseng et al. also describes a collaborative drawing annotation system for augmented reality with a focus on perspective preservation [28]. In their annotation system, users can view a history of how a 3D freehand-drawn annotation was created over time. In addition, an avatar shows the annotator's position and orientation during the process of creating the annotation.

DESIGN CONSIDERATIONS

While designing an annotation system specifically for VR and 360º environments, I first needed to define the essential parts of an annotation and how these parts might be designed in a 3D environment. Four essential parts were defined in my explorations; 1) Space, 2) Point of interest, 3) Link, and 4) Content (Figure 2).

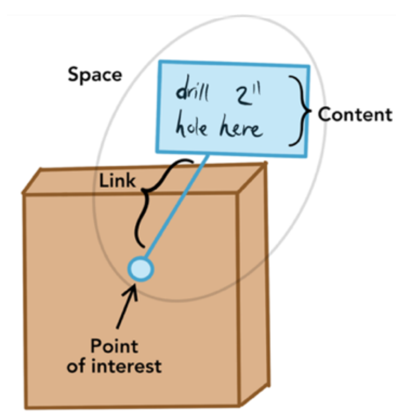

Figure 2. Annotation on a box in 3D with parts labeled

Space

Space refers to the physical area that the annotation takes up. For example, the space of an annotation may be the boundaries of the text box for an annotation or a yellow post-it note on a whiteboard. In a physical world, space can be a limiting factor when annotating a physical document. However, the digital world allows us to free up space by hiding, limiting the view of, and recalling annotations only when the user wants to view them. As a result, the VR world could potentially benefit greatly from a system that manipulates the space that an annotation takes up. Like annotation systems in 2D digital documents, an annotation system in VR should be able to toggle between showing and hiding different annotations.

Point of Interest

Point of interest refers to the subject or subjects that are being annotated. Identifying the point of interest is the first step in creating an annotation. As a result, the process of doing so must feel intuitive. Without the point of interest, the annotation no longer has an anchor attaching itself to the virtual world and thus loses context.

Link

Link refers to the connection between the annotation content and the point of interest. The link may be explicitly portrayed (e.g. by using a line to point from the annotation to point of interest), or it may be inferred through other visual cues or user interactions with the annotation.

Content

Content refers to the actual annotation itself and what information is stored. Most commonly, the annotation content includes text and possibly even metaobjects like arrows or circles. The digital nature of VR allows for various types of multimedia annotations, including audio, video, graphics, and even portals to other experiences.

Considered Factors

While creating an annotation system in a virtual 360º environment, there were several factors that I considered important throughout my design process.

Context-Preserving

While researching other annotation systems in VR, I found that most annotations do not capture the context of where and how an annotation was created, and this information can be crucial for understanding the content of an annotation. For example, a simple audio annotation only plays a voice recording, but this does not give any additional information about where the annotator was looking and even positioned in the environment. As a result, I focused on designing ways to preserve as much about the annotator's context and incorporating that into the annotation.

Consistency

Another issue with annotation systems in virtual 360º environments is consistency during annotation viewing and playback. This is extremely evident in a 2D metaobject annotation, such as a circle, that is simply placed in the world without considering other viewing angles. If the 2D circle is not viewed from the correct angle in VR, it can be difficult to determine where the circle is placed in relation to other objects—from the side, a 2D circle may look like a simple line in 3D space. Thus, I aimed to design an annotation system that promotes consistency in the annotation viewing and playback experience.

Reducing Movement

Through the inputs of a VR headset and controllers that track movement and orientation, we can design features in VR that require the user to physically move and/or rotate their hands, body, and head. While features that require movement can create a more immersive experience, these features can often become physically tiring after extended use. Because users will often need to create multiple annotations in one VR session, I aimed to design an annotation system that limits the amount of unnecessary movement of the controllers and headset to ensure a more comfortable annotation experience.

Intuitive

The annotation process should feel straightforward to the user, no matter what type of annotation the user is creating. Because VR is a relatively new medium for most users, the annotation experience should be as intuitive as possible for even beginner users, while still considering the other factors listed before.

VR-NOTES

After considering these factors, I created VR-Notes, a design for a multimedia annotation system in VR that captures information about the annotator's perspective and context, later displaying this information to provide a richer and clearer annotation viewing experience. There are three types of annotations in this system—doodle annotations, audio annotations, and text annotations.

Annotation Creation Process

Creating each annotation begins with the same process. First, the user selects a point of interest in the 3D environment using the main trigger button and a virtual laser pointer (Figure 3a). After selecting a point of interest, a menu appears above the left controller, allowing the user to then select what type of annotation they want to create (Figure 3b). The user can also change the annotation's point of interest by selecting another location using the main trigger and virtual laser pointer. The next steps vary by annotation type, but the user can cancel out of the annotation process by pushing a highlighted "X" button on the right controller.

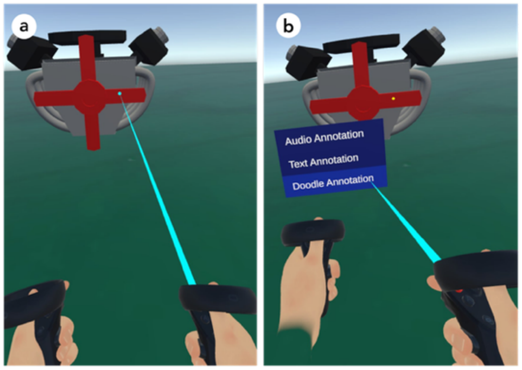

Figure 3. (a) The user starts annotating by first selecting a point of interest using the laser controller. (b) The user then selects an annotation type on a menu that appears above the secondary controller using the laser.

Doodle Annotations

After selecting a point of interest, the 2D clear plane with a drawing grid and border appears between the user and the point of interest. The plane represents the drawing space for the user's doodle annotation, and the user draws on the plane by holding down the main trigger and moving the laser pointer (Figure 4). On the side of the plane, the user can use the CLEAR button to clear the plane and delete the doodles, or the SAVE button to save the annotation. If the annotator moves around, the clear plane will move to stay between the annotator and point of interest, as well as rotate to face the annotator so that it remains flat.

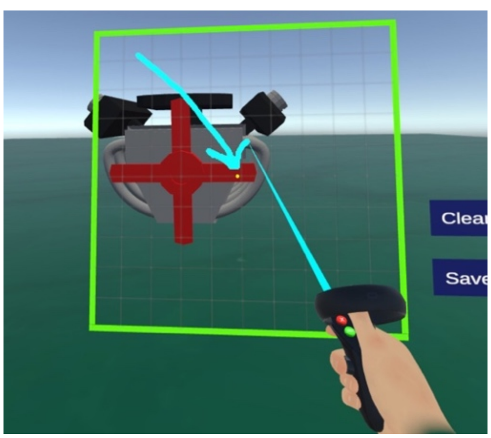

Figure 4. In a "doodle" annotation, a drawing canvas appears between the user and point of interest, shown here with an outline in bright green. The canvas has a square grid, and the user can draw on it with the laser.

After saving the annotation, a projection of the clear plane and the background from the user's point of view is captured and saved as the doodle annotation "image." This planar drawing space allows users to comfortably draw on a 2D space that aims to feel more natural than a freehand 3D drawing. Additionally, by capturing the perspective of the annotator, VR-Notes aims to provide more context about how the annotation was meant to be viewed.

Doodle Annotations Alternatives

In designing a doodle annotation system in VR, there are several options which I could have pursued.

Freehand Drawing in 3D

A freehand drawing method could be created to mirror the drawing systems of Tiltbrush and Quill. Although this method might be the most popular method in the VR community, there are several reasons why I did not select it. Many users are still accustomed to annotations in the two-dimensional world—either on a physical piece of paper or digital document. Especially considering how our natural reading and writing skills are commonly developed in a 2D medium, drawing in 3D can still be disorientating and hard to interpret for other users [4, 5, 8]. Additionally, there is no information about the annotator's perspective while viewing freehand drawings at a later time.

Drawing on Model and Environment

Another method design could allow the user to draw directly on the texture of the model or environment itself. This concept of lasering or painting on surfaces is easy to understand and for new users to learn. However, this method restricts the format of how annotations can be drawn. For example, a user cannot draw in the gaps of an object, like the space between the legs of a table. Users are required to draw directly on the object, which can be also hard to interpret if the object's surface is irregular or curved.

Audio Annotations

After selecting a point of interest, the user can start an audio recording by pressing the record button highlighted on the controller (Figure 5a). Using the microphone in the headset, audio from the user is recorded. Additionally, the system records position and orientation information of the user's head anchor and controllers. Red dots blink around the controllers to remind the user that audio and motion is being recorded. The user can also use the main triggers on either controller as laser pointers, and effectively point out certain details on the object of interest during the recording. When the user is done recording the audio annotation, the user clicks the stop button highlighted on the controller, and the audio annotation is saved (Figure 5b).

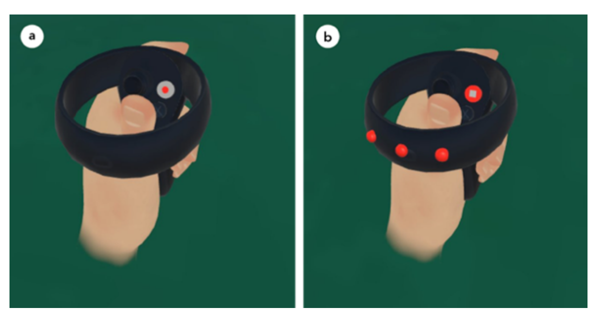

Figure 5. (a) Record button to start audio annotation. (b) Stop button to stop recording, and also blinking red lights around the controller to show an active recording session

Text Annotations

While doodle and audio annotations are the main focuses of VR-Notes, I also developed a simple text annotation feature. For text annotations, a virtual QWERTY keyboard (asset from VR UIKit) appears between the user and the point of interest. The virtual keyboard moves with the user to stay in between the user and the point of interest and also rotates to directly face the user, much like the doodle plane. The user types on the keyboard with virtual pointers from the controller and the physical main trigger buttons. Clicking the ENTER key on the keyboard will save the annotation.

Annotation Viewing Process

When an annotation is saved, an annotation marker at the location of the point of interest is created. Users can then select the annotation marker and focus on it using the main trigger, which allows the user to view the annotation.

Viewing Doodle Annotation

When focusing on a doodle annotation, a display appears above the left controller, showing the image capture of the doodle and background. This display is anchored to the left controller, so the viewer can move the display to a comfortable viewing angle and position. With the background perspective preserved in the annotation, the user can view the doodle with context of the viewing perspective of the original annotator (Figure 6b).

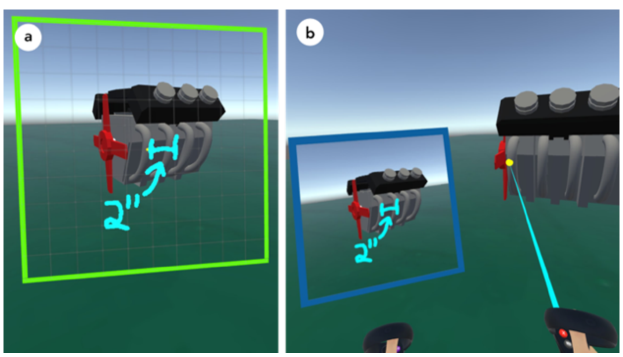

Figure 6. The annotator's perspective (a) is preserved as the background of the annotation when the annotation is viewed at a later time (b).

Viewing Audio Annotation

When focusing on an audio annotation, the audio recording begins to play. During playback, an avatar consisting of models of a head and two controllers are animated according to the original annotator's position and orientation values during the audio recording (Figure 7). Additionally, information about the laser pointers being turned on and off carries over into the playback. The audio is played from the position of the head of the avatar, and using an audio spatializer made for Oculus VR, the sound comes from the direction of the avatar in the environment. The user can also click a restart button highlighted on the controller, which replays the audio annotation from the beginning.

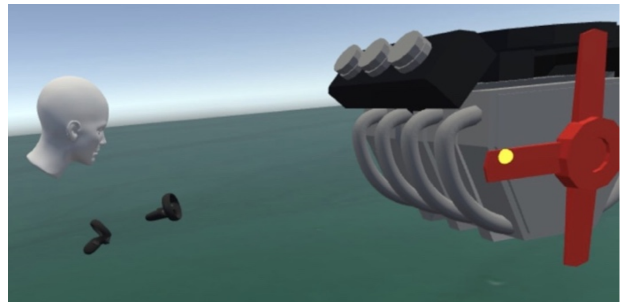

Figure 7. Animated avatar during audio annotation playback

Viewing Text Annotation

When focusing on a text annotation, a display appears above the left controller, showing the text of the annotation. Like the doodle annotation display, this display is anchored to the left controller, so the viewer can move the display to a comfortable viewing angle and position.

Other Viewing Features and Functionality

There were also several other features in VR-Notes aimed at benefiting the annotation viewing experience and improving the overall user experience.

Teleportation

While focusing on an annotation, a user can click on a teleport button highlighted on the controller (Figure 8a), and teleport to the location of where the annotation was created (using a fade in/fade out transition to prevent nausea). This functionality allows viewers to immediately teleport to the original annotator's location to easily see the point of interest from the annotator's perspective.

Delete

Users can also delete annotations by focusing on the annotation and then holding the delete button highlighted on the controller (Figure 8d). Deleted annotations will not be included in any exports.

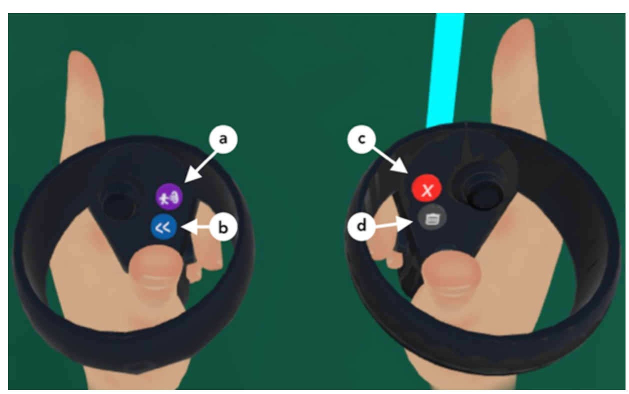

Figure 8. Options while viewing annotation—(a) Teleport to annotator's position/orientation, (b) Replay (only audio annotations), (c) Stop annotations focus, (d) Hold to delete annotation

Hide/Show Annotations Toggle
From the main menu, users can toggle between showing and hiding all annotations in the VR environment.

Library View
From the main menu, users can also select a mode to view the library view, a single menu of all of the annotations in the local environment (Figure 9). This view provides previews of the annotations that have been created in this environment. By clicking on an annotation in the library view, the user will be teleported to the location where the annotation was created and also focus the annotation for viewing.

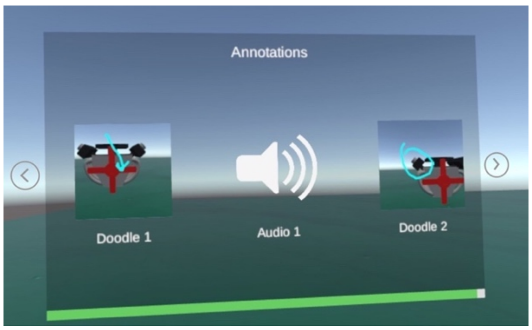

Figure 9. Library view of all annotations in environment

Export and Import
Users can export their annotations to the local storage of the headset, as well as import data from local storage in the main menu settings. This data is stored in a session specific folder structure.

Not Considered
In VR-Notes, I am not considering the various ways that we can input text for text annotations, including voice-to-text. Additionally, I am not considering what is the best navigation method for this type of annotation system, and I am using the default joystick navigation controls provided with the Oculus SDK.

IMPLEMENTATION
The following section outlines the hardware and software implementation architecture for VR-Notes.

Hardware
For hardware, I chose to work with the Oculus Quest because of its untethered functionality that allows it to be used without a powerful gaming PC. The Quest comes with two wireless controllers, which allow for button and joystick inputs. Additionally, the Quest provides six degrees of freedom, which allows the headset to track and relay position and orientation information from the headset and controllers

Software
Working with the Oculus Quest, I chose to use Unity and the Oculus Developer SDK to build my prototype, as these are popular platforms for development in this space. While working in Unity, my primary coding language for this project was C#. The Oculus Developer SDK includes several frameworks to set up player navigation and controlling, which laid the foundation for the user interactions in my prototype.

Additional Assets Used
To build out additional features, I used the VR UIKit Asset from the Unity Store for the keyboard input in text annotations, as well as the menu layout for the library view feature.

CODE ARCHITECTURE
The software was implemented using Unity (C#) along with the Oculus SDK.

Creating Annotations
To select a point of interest, a Raycast from the controller position is created and directed in the forward direction of the controller. Upon a hit with an object's collider, the Vector3 position of the hit is returned, as well as the hit object. This system is used for all interactions with the virtual laser pointers in my prototype. On a valid hit of a point of interest, a menu prefab appears, and each menu option calls its own function to begin the respective annotation's process.

Doodle Annotations
In doodle annotations, a clear plane is created between the user and point of interest. While the plane is active, it is constantly updating to position itself between the user and point of interest, as well as rotating to face the user. Using the Raycast hit method, the system is able to identify exactly which pixel of the clear plane texture should be colored for the doodle. The system decides to color which pixels through a line drawing algorithm in the plane texture. If the CLEAR button is pressed, the clear plane texture is reset.

To capture the background of the clear plane from the user's perspective, a separate Camera object is anchored to the user's head anchor. A script adjusts the Camera's field of view and perspective such that its view is confined to the area outlined by the clear plane. When the doodle annotation is saved, the Camera captures a screenshot of this background and blends it with the doodle clear plane to create the doodle annotation capture. While viewing a doodle annotation, the image of the doodle capture is pulled from the storage and updated to the doodle display.

Audio Annotations
In audio annotations, a script captures audio from the headset's built-in microphone, creating an AudioSource object to save as the recording. Additionally, information about the position and orientation of the head and controller anchors are stored as Vector3's. This information is collected every 0.25 seconds, as well as if the lasers are turned on during that time. During audio annotation

playback, an avatar consisting of a head and two controller models is activated. The AudioSource begins playback from the avatar head location using the Oculus Audio Spatializer, and the avatar position and orientations are lerped between the saved values to represent the original annotator's movements.

Text Annotations

In text annotations, the keyboard prefab is generated along with the input field. The keyboard updates its position to be between the user and point of interest, as well as rotating to face the user. While viewing a text annotation, the string of the annotation is retrieved from the storage and updated to the text display.

Export and Import

Using a Stream Reader and Stream Writer, the system can export and import data from the local storage of the Oculus Quest. Position and orientation information is stored as Vector3's and written to a CSV file. Doodle image captures are stored as PNG files, and audio recordings are stored as MP3 files.

USER STUDY: UNDERSTANDING DRAWING WITH DOODLE METHOD

The goal of this study was to compare participant usage of the VR-Notes doodle method with the freeform drawing method as the control method. I chose to implement the freeform drawing method as the control method because this method mirrors the popular drawing technique in popular commercial VR applications like Tiltbrush and Quill, and this drawing system can easily be used as an annotation drawing tool.

Participants:

Because of the circumstances surrounding COVID-19 and social distancing recommendations, I was not able to conduct in-person experiments with a sufficient participant pool. For the time being, I have been able to run the experiment with three people living in my household. All three participants (2 females, 1 male) were right-handed, aged 22 to 57. Two of the participants had little to no VR experience before the experiment, and one participant had proficient experience developing for VR. Although all of these individuals lived with me and knew I was doing research in the VR space, only one of them knew which doodle method I designed.

Apparatus:

The study was conducted in a large empty room using an Oculus Quest headset and Oculus Touch controllers that came with the headset. During the study, the positions and orientations of the user's headset and controllers were tracked. Likewise, various timers kept track of how long it took users to complete certain tasks. Annotation data was also recorded to later analyze accuracy. All of this information was exported to the Oculus Quest's local storage after completing an experiment trial.

Task and Procedure:

The task required participants to circle a highlighted target or group of targets in a VR scene using the annotation system being tested. In the VR scene, there was a "cart" containing several different objects, and these objects were either spheres or cylinders (to represent two different types of target shapes). Users were told to circle the highlighted object as clearly as possible (Figure 10a-b), and that other users would need to be able to determine which object was circled without the object being highlighted. Participants were allowed to move to a different location or perspective to create their ideal circle annotation. While drawing the circle around the highlighted object, the user can clear the annotation and start over by pressing a physical button on the left controller. Once the user was done, the user was asked to confirm the annotation by pressing a button on the right controller. After confirming the annotation, a new object was highlighted, and the user was told to repeat the annotation process with the new object.

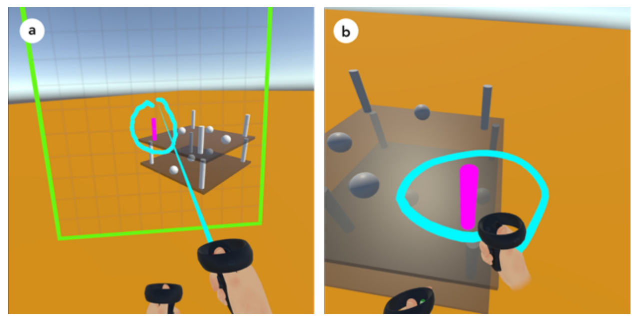

Figure 10. (a) Experiment method: VR-Notes doodle annotation, (b) Control method: freehand 3D drawing method

Before the experiment began, there was a tutorial session lasting 10-15 minutes that taught the user how to use the current navigation system. Participants were also taught during this time how to use the two different annotation systems and were allowed to complete a "mock" round for both methods. Between rounds, users were also able to take a break before continuing on.

After the interactive experiment session, the participants were asked about their experience with the annotation system based on specific qualities. They responded using a 7-point Likert scale questionnaire (1: strongly disagree to 7: strongly agree) to the following statements.

- *Ease of use*: "The annotation tool was easy to use and intuitive."
- *Accuracy*: "The annotation tool allowed me to be accurate in circling."
- *Enjoyment*: "I enjoyed the annotation process using this annotation tool."
- *Considering others' perspectives*: "Other people can tell which objects I circled when viewing my annotations."

Participants were also asked why they gave a certain score. Additionally, they were asked which annotation system they would prefer to use in a productive work setting. Finally, users were prompted for any other questions or comments.

Study Design

The experiment employed a 2x2x2 within subject factorial design. The independent variables were:

Method **(**VR-Notes doodle method/control method)

Complexity **(**Simple/complex): The cart contained either a few objects or many objects. This aimed to test how each annotation system would perform in a crowded space versus an open space.

Number of targets (Single/multiple): Each trial had either a single highlighted target object or a group of three highlighted target objects, and this aimed to test how each annotation system would perform at circling single objects compared to a group of objects.

See Figure 11 for the differences in *Complexity* and *Number of targets* in the carts.

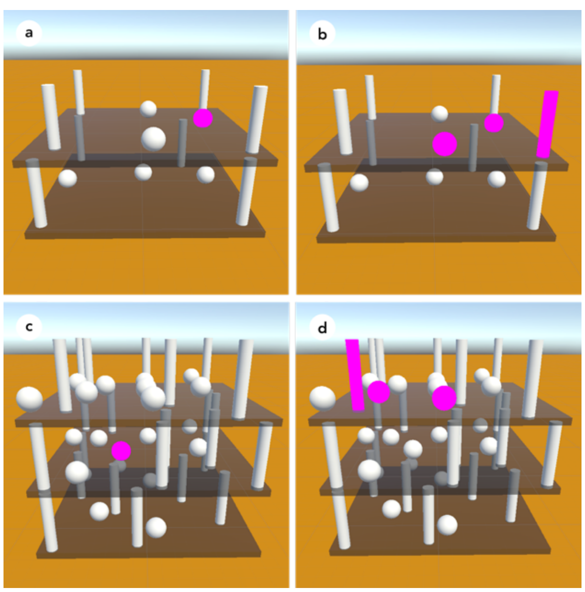

Figure 11. (a) Simple cart - single target, (b) Simple cart - multiple targets, (c) Complex cart - single target, (d) Complex cart - multiple targets

Each round contained 10 trials of annotations, totaling 80 trials per participant. The highlighted objects were selected randomly during experiment planning and were kept the same between the VR-Notes doodle method and control method rounds. Additionally, these highlighted objects were kept the same for all user participants. Half the users started with the VR-Notes doodle method, and the other half started with the control method.

Results and Analysis

After exporting CSV, I parsed the data using Python scripts and libraries. Then, I exported the data to SPSS to analyze through paired t-tests and ANOVA tests.

Movement

Movement was calculated over time from the various data points using the distance formula. Distance was measured in meters through Unity's world scale settings. For each experiment round of ten annotation trials, the sum of the movement distance was calculated.

Through a paired t-test, I found a significant effect of *Method* on head movement ($t(11) = 4.228$, $p = 0.001$), left controller movement ($t(11) = 3.291$, $p = < 0.01$), and right controller movement ($t(11) = 4.513$, $p = 0.001$). ANOVA yielded significant interactions effects on *Method* x *Number of targets* for head movement ($F_{1,24} = 9.922$, $p < 0.01$), left controller movement ($F_{1,24} = 7.044$, $p < 0.05$), and right controller movement ($F_{1,24} = 22.620$, $p < 0.001$).

Participants moved their head and two controllers less on average during the VR-Notes doodle method when compared to the control method (Figure 12). Head movement averaged 6.397 meters (s.e. 0.676) for the VR-Notes doodle method and 13.651 meters per round (s.e. 1.812) for the control method. Left controller movement averaged 8.044 meters per round (s.e. 0.608) and 14.345 meters per round (s.e. 2.063), while right controller movement averaged 14.757 meters per round (s.e. 0.780 and 30.028 meters per round (s.e. 3.565) for the VR-Notes doodle method and control method respectively. When circling multiple target objects compared to a single target object, the data shows a significant increase in movement for the head and two controllers for the control method, but little change for the VR-Notes doodle method.

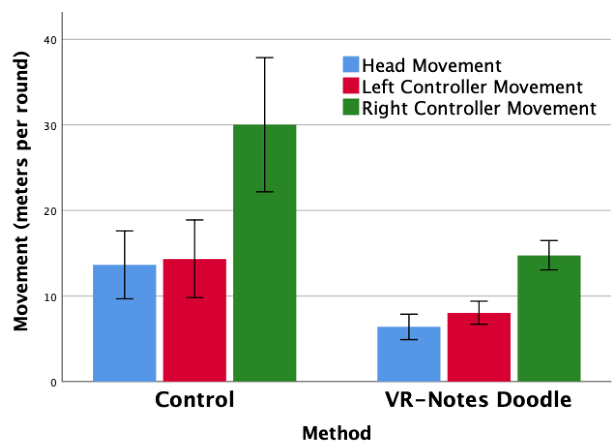

Figure 12. Average movement per round for head, left controller, and right controller by method (error bars show 95% CI in all figures)

Rotation

Rotation was calculated over time by finding the distance between the rotation data for an anchor point. Rotations are recorded using unit quaternions, and the distance is a positive scalar corresponding to the chord of the shortest path/arc that connects the two quaternions accounting for sign ambiguity. For each experiment round of ten annotation trials, the sum of the rotation distances was calculated.

Through a paired t-test, I found a significant effect of *Method* on head rotation ($t(11) = 2.736$, $p < 0.05$), left controller rotation ($t(11) = 2.540$, $p < 0.05$), and right controller rotation ($t(11) = 3.927$, $p < 0.005$). ANOVA yielded significant interactions effects on *Method x Number of Targets* for head rotation ($F_{1,24} = 4.499$, $p < 0.05$) and right controller rotation ($F_{1,24} = 13.623$, $p = 0.001$).

When annotating using the VR-Notes doodle method, participants rotated their head and controllers less when compared to the control method (Figure 13). During a round of annotation trials, head rotation averaged 12.133 rotation distance per round (s.e. 1.006) for the VR-Notes doodle method and 20.815 rotation distance per round (s.e. 3.454) for the control method. Left controller rotation averaged 13.026 rotation distance per round (s.e. 0.697) and 22.051 rotation distance per round (s.e. 3.720), and right controller rotation averaged 19.983 rotation distance per round (s.e. 1.202) and 36.239 rotation distance per round (s.e. 4.068) for the VR-Notes doodle method and control method respectively. When circling multiple target objects compared to a single target object, the data shows a significant increase in rotation for the head and right controller for the control method, with only a small increase for the VR-Notes doodle method.

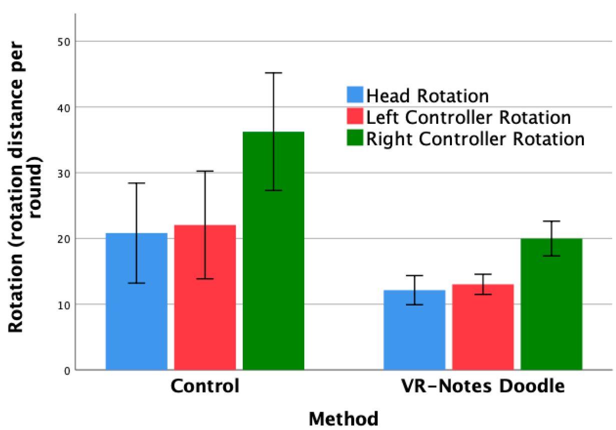

Figure 13. Average rotation per round for head, left controller, and right controller by method

Time

Through a paired t-test, I found a significant effect of *Method* on time spent drawing the annotation ($t(118) = 2.008$, $p < 0.05$) and total time to complete annotation ($t(118) = 3.543$, $p = 0.001$). Three outliers were identified using an SPSS stem and leaf plot and removed from the data.

On average, the VR-Notes doodle method took longer (7.543 seconds, s.e. 0.588) than the control method (5.067 seconds, s.e. 0.548) to complete each annotation. However, users spent less time drawing in the VR-Notes doodle method trials (2.899 seconds, s.e. 0.161) than the control method trials (3.276 seconds, s.e. 0.231).

Accuracy (Center of Annotation)

As a measure of accuracy, I modeled a best-fitting ellipse of the points in the circle annotation and compared the center of the ellipse to the true center of the target object. For annotations with multiple target objects, I compared the center of the ellipse to an average center of the group of targets.

For the VR-Notes doodle method, the circle annotation exists on a 2D plane, so I used a total least squares estimator to generate a model of a best-fitting ellipse model (Figure 14). After locating the center of the target object(s), I then found the distance between the ellipse center and true target center. For the freehand drawing method, the circle annotation and target locations exist in 3D space. As a result, I first found a least square, best-fitting plane through the set of 3D points representing the circle annotation. After projecting the 3D annotation data points onto this best-fitting plane, I then used the same total least squares estimator to model the ellipse of the projected circle annotation (Figure 15). I was then able to compare the center of the circle annotation in 3D space with the target(s) center.

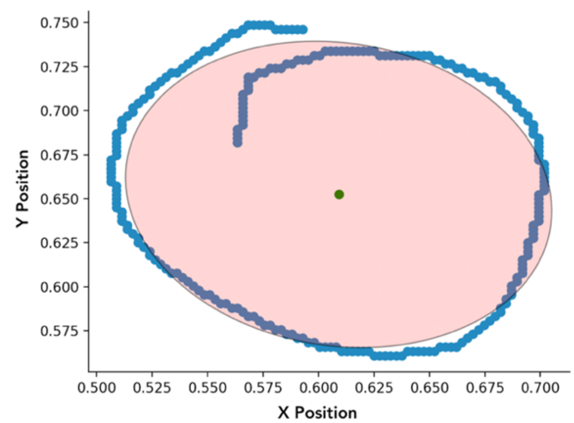

Figure 14. TLS best-fit ellipse (red ellipse) from the annotation data (blue points) and calculated center (green)

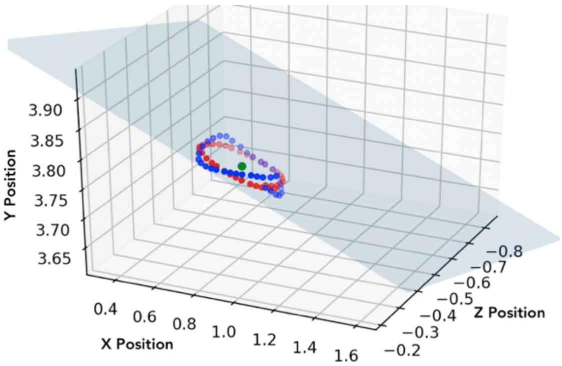

Figure 15. Best-fit plane from annotation data (blue points), Projected points onto plane (red points), calculated center (green point) of TLS best-fit ellipse of projected annotation

Through a paired t-test, I found a significant effect of *Method* on distance to true center ($t(119) = 13.492$, $p < 0.001$). On average, user's annotations were closer to the target center when using the VR-Notes doodle method (0.0217 meters, s.e. 0.002) compared to the control method (0.0643 meters, s.e. 0.004).

User Feedback Scores

In summary, the participants scored the VR-Notes doodle method higher than the control method in all four areas (Figure 16). Participants strongly agreed that the VR-Notes doodle method was "easy to use" (7.00, s.e. 0, with 7 being strongly agree), and gave a lower score for the control method (5.67, s.e. 0.882). Participants also agreed that they were accurate using the VR-Notes doodle method (5.67, s.e. 0.882) and gave the control method a lower score (4.33, s.e. 0.333). Additionally, participants strongly agreed that they enjoyed using the VR-Notes doodle method (6.67, s.e. 0.333) over the control method (3.67, s.e. 0.667). Finally, participants agreed that others could interpret their annotation correctly on the VR-Notes doodle method (6.00, s.e. 0), while less confident for the control method (4.00, s.e. 0.577).

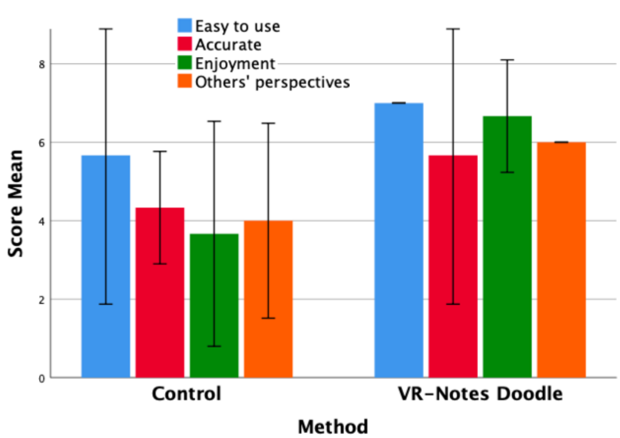

Figure 16. Average rating for each statement by method

User Feedback Comments

All user participants preferred to use the VR-Notes doodle method over the control method in a work-setting. Additional comments are detailed below.

"Easy and Intuitive"

Two participants both said that it was easier and more natural to draw on a 2D surface rather than in 3D. One participant mentioned how both methods were easy to pick up, but the control method required much more physical movement. Two participants also mentioned that they were tired after the control method.

"Accurate"

One participant said that the control method was less accurate because they had difficulty drawing in 3D space. One participant mentioned that it was difficult to get the right perspective for the VR-Notes doodle method. All participants mentioned how they were more accurate when circling a single object versus a group of objects.

"Enjoyment"

Two participants said that both methods were fun to use during the single objects, but the control method was more physical work on the groups of objects. One participant enjoyed the VR-Notes doodle method because they were able to stand in one spot when drawing without having to move around too much.

"Considering Others' Perspectives"

All three participants believed that annotations for both methods would be more easily interpreted in simple and single rounds over the complex and multiple target rounds. Two people were concerned that the control method annotations would be hard to comprehend if other people viewed it from the wrong angle.

Summary

The early study suggests certain benefits of utilizing the VR-Notes doodle method over freehand drawing in 3D space. Primarily, the VR-Notes doodle method reduced movement by 53%, 44%, 51% and reduced rotation by 42%, 41%, 45% for the head, left controller, and right controller respectively. Additionally, annotating multiple objects using the control method required an increase in movement and rotation of the head and two controllers, but no significant increase was found for the VR-Notes doodle method. While users may spend more time in the annotation process for the VR-Notes doodle method (which participants attributed to finding the right perspective of the target object), users spent 13% less time on the drawing portion of the annotation process for the VR-Notes doodle method. The experiment also suggests that users are more accurate in centering their annotation using the VR-Notes doodle method over freehand drawing, as participants were more accurate in drawing on a 2D plane compared to an open 3D space. This finding supports previous research showing the challenges that people face when drawing in a 3D space [5, 9]. Ultimately, there would need to be additional trials with more participants to validate these early results.

DEMO APPLICATIONS

The inspiration for my research came from seeing the innovative VR work of professors across Dartmouth College. Throughout my design process, I have closely worked with two of these professors—Nicola Camerlenghi of the Art History Department and Jonathan Chipman of the Geography Department. Both of the professors have expressed the need for an annotation system in their VR work. While their specific needs may be different, they provided valuable feedback for my designs and ideas. Collaborating with these two professors, I was given 3D models from their work to create two demo applications for VR-Notes.

Tour - Saint Paul's Outside the Walls (1823)

Camerlenghi has continuously worked on recreating the Saint Paul's Outside the Walls church that burned down in 1823 in 3D modeling technology. He believes that VR will be an innovative way to experience the church's lost architecture and artwork. By using VR technologies, he also hopes to create educational experiences from virtual tours of the church, where annotations will lead users through the church. His specific use cases include identifying specific details on artwork and architecture, point out distances between architecture features, and tell a narrative of the church through the annotation system.

Before using VR-Notes, Camerlenghi would need to take a screen and voice recording of the VR experience, and then edit this experience for later viewing. In the recording, he could use a virtual laser pointer to point out specific features while in VR, but none of this information would be stored in the virtual environment itself. If the students were able to download the application to their own VR headset, they would still need to constantly take the headset off and put it on again to find which parts of the environment to explore.

With VR-Notes, Camerlenghi can now highlight points of interest in the architecture and artwork directly in the world itself (Figure 17). He can leave narratives and audio recordings for students to listen to, and also use the virtual laser pointers during these audio recordings to point out specific details. Now, students can download the application with Camerlenghi's annotations and take full advantage of the immersive qualities of VR without having to constantly take the headset off. Students can also respond to prompts or questions by adding their own annotations and export these annotations so they can be reviewed by Camerlenghi.

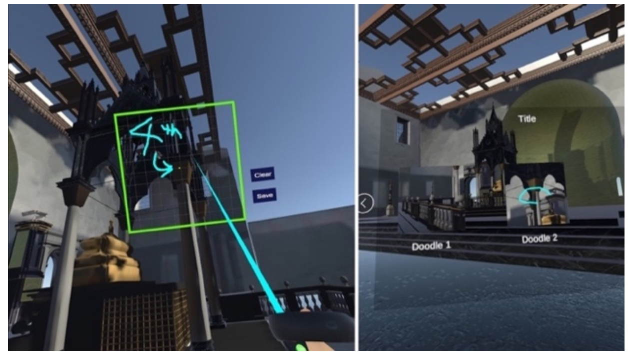

Figure 17. Demo of annotation system in early models of Saint Paul's Outside the Walls in VR

Field Work - Grand Canyon

Chipman has begun exploring how he might use VR to make field work more accessible for all students. Currently, conducting field work is an important skill in the introductory earth sciences classes at Dartmouth. Students are expected to go on a field trip to explore the concepts they are learning in the classroom in the actual physical world. This can be extremely challenging for students who are physically disabled, and these students are often given an alternative assignment that involves writing a paper or watching a video. A VR application could be a potential solution to accommodate the needs of physically disabled students while giving them a more realistic experience of field work.

Additionally, many geographic locations of interest to researchers are very sensitive to human interaction. Other regions, like lava beds and areas with quicksand, can be extremely dangerous. Some places, like the surface of Mars, are not even reachable yet. However, modeling software can recreate these environments using photogrammetry data. By doing so, we can recreate the environments in VR and allow researchers to conduct virtual field work.

Chipman's use cases include allowing students to find information about specific aspects of the environment, having them respond to prompts of questions, and finally taking measurements of distances and angles. Before using VR-Notes, Chipman's students can enter the VR environment, but aren't able to interact with the environment in a meaningful way. Instead, they need to constantly take the headset off and watch accompanying videos or read reference materials to learn about the environment they are in.

With VR-Notes, Chipman will be able to leave annotations for his students to interact with and learn more about the environment (Figure 18). His students would be able to respond to prompts of questions directly in VR itself and export this information to be reviewed. With further application-specific feature development, his students will be able to take distance and angle measurements of the environment.

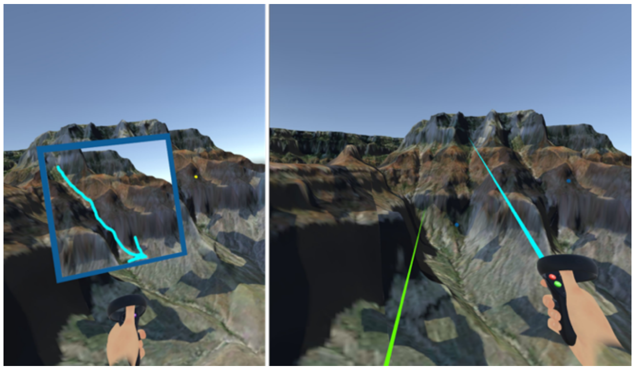

Figure 18. Demo using annotation system to complete field work in Grand Canyon model in VR

DISCUSSION AND LIMITATIONS

Throughout the research process, there are many learnings and next steps that I want to discuss in this section, as well as some limitations.

Data Export and Import

For my prototype, I was able to implement features for exporting and importing data through the local storage. However, I understand that this is not the most effective way for users to manage the data. My design for an annotation system would greatly benefit from the addition of a more streamlined process, possibly utilizing syncing technologies and cloud services directly within the VR application and uploading this information into an intuitive online platform to interact with the data.

Cross-Hardware Concerns

While I chose the Oculus Quest headset for my prototype and experiments, there are various different VR headsets available to the public at this time. These headsets vary in different ways, including processing power, controller design, and additional features. My design assumed that the headset can track six degrees of motion (the position and rotation of the headset and controllers in space). Yet, this is not the case for all headsets on the market today. Due to this, my design may need to be changed for different types of headsets and controllers depending on their feature set. Likewise, I assumed that a VR headset used in this productivity context will come with two controllers. However, future headsets may not even include physical controllers, instead opting for hand-tracking (see Oculus Hand Tracking [2]). As a result, these interactions will need to be reconsidered based on the availability of user inputs.

Text Input

As mentioned in the design consideration, I did not explore methods to input text in VR. However, I believe that this is a worthy area to explore that would benefit the user experience in any annotation system that uses text input. The professors that I worked with and others interested in VR technologies have mentioned the difficulty of typing on a floating digital keyboard in VR using virtual pointers. Voice input serves as a better alternative in some use cases, but there needs to be more exploration into effective text input systems in VR.

Collaboration

Users would benefit from the ability to sync and update annotation projects automatically. Ideally, one user would leave annotations in the world on their headset, and these annotations would be synced via cloud services. When another user enters the world, they now can see the new annotations and do not need to worry about importing the most current version with up-to-date annotations. Additionally, there could be possible explorations into the opportunities and challenges of a live networking experience utilizing this annotation system.

Application-Specific Features

Depending on the specific user scenario, there would need to be the development of additional features and annotation tools to ensure maximum productivity in the VR space. For example, the field work demo would benefit greatly from a tool that allowed users to make measurements of angles and distances in space. Additionally, users may benefit from being able to insert images or videos from local storage onto a doodle annotation. Further development of different drawing features could also be helpful, including color pickers, canvas resizing and customization, and brush type.

CONCLUSION

I introduced a design for an annotation system in VR that captures context from the annotator's perspective and presents the annotations with this perspective information. To prototype the design, I created a proof-of-concept system VR-Notes using an Oculus Quest, Oculus SDK and Unity. I explored several different annotation methods and implemented a number of features and demo applications to illustrate the user experience. Through an early experiment, the results suggest that the VR-Notes doodle method reduced user movement and rotation during the annotation process, improved centering accuracy, and reduced the time spent drawing during the annotation process when compared to a popular 3D freehand drawing method. Users were optimistic about the potential of the VR-Notes annotation system design and scored it higher on various qualities over the control method. Future work will focus on building application specific features for the annotation system, as well as more streamlined export and import processes with annotation data. The addition of these features will aim to increase user productivity and efficiency while using the annotation system.

ACKNOWLEDGEMENTS

I would like to thank Xing-Dong Yang for advising me and providing feedback throughout my research project in the

HCI field. I would also like to thank James Mahoney for introducing me to the world of VR/AR and showing me how to bring ideas to life in this area. Finally, I would like to thank Nicola Camerlenghi and Jonathan Chipman for inspiring my ideas for a research project through their own work with VR and providing feedback throughout my design process.

REFERENCES

[1] 2016. Quill. Retrieved May 11, 2020 from http://quill.fb.com.

[2] 2019. Thumbs Up: Hand Tracking on Oculus Quest This Week. *Oculus Blog*. Retrieved May 11, 2020 from https://www.oculus.com/blog/thumbs-up-hand-tracking-now-available-on-oculus-quest/.

[3] 2016. Tiltbrush. Retrieved May 11, 2020 from http://tiltbrush.com.

[4] Wadee Alhalabi. 2016. Virtual reality systems enhance students' achievements in engineering education. In *Virtual Reality in Learning, Collaboration and Behavior*, 912-925.

[5] Claudia Armbrüster, Marc Wolter, Torseten Kuhlen, W. Spijkers, Bruno Fimm. 2008. Depth Perception in Virtual Reality: Distance Estimations in Peri- and Extrapersonal Space. In *CyberPsychology & Behavior.* Volume 11, 9-15.

[6] Rahul Arora, Rubaiat Habib Kazi, Frazer Anderson, Tovi Grossman, Karan Signh, George Fitzmaurice. 2017. Experimental Evaluation of Sketching on Surfaces in VR. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, (CHI '17), 5643-5654.

[7] Mayra D. Barrera Machuca, Paul Asente, Jingwan Lu, Byungmoon Kim, Wolfgang Stuerzlinger. 2017. Multiplanes: Assisted Freehand VR Drawing. In *Adjust Publication of the 30th Annual ACM Symposium on User Interface Software and Technology* (UIST '17), 1-3.

[8] Mayra D. Barrera Machuca, Wolfgang Stuerzlinger, Paul Asente. 2019. Smart3DGuides: Making Unconstrained Immersive 3D Drawing More Accurate. In *Proceedings of ACM Symposium on Virtual Reality Software and Technology* (VRST '19), 1-13.

[9] Mayra D. Barrera Machuca, Wolfgang Stuerzlinger, Paul Asente. 2019. The Effect of Spatial Ability on Immersive 3D Drawing. In *Proceedings of the 2019 Conference on Creativity and Cognition* (C&C '19).

[10] A.J. Bernheim Brush. 2002. Annotating digital documents: anchoring, educational use, and notification. In *CHI '02 Extended Abstracts on Human Factors in Computing Systems* (CHI '02), 542-543.

[11] A.J. Bernheim Brush, David Bargeron, Anoop Gupta, J.J. Cadiz. 2001. Robust annotation positioning in digital documents. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (CHI '01), 285-292.

[12] J.J. Cadiz, Anop Gupta, Jonathan Grudin. 2000. Using Web annotations for asynchronous collaboration around documents. In *Proceedings of the 2000 ACM conference on Computer supported cooperative work* (CSCW '00), 309-318.

[13] Damien Clergeaud, Pascal Guitton. 2017. Design of an annotation system for taking notes in virtual reality. In Proceedings of the *2017 3DTV Conference: The True Vision - Capture, Transmission and Display of 3D Video* (3DTV-CON '17), 1-4.

[14] Joshua Q. Coburn, Ian Freeman, John L. Salmon. 2017. A Review of the Capabilities of Current Low-Cost Virtual Reality Technology and Its Potential to Enhance the Design Process. In *Journal of Computer and Information Science in Engineering,* 17(3).

[15] Steven Feiner, Blair MacIntyre, Dorée Seligmann. 1992. Annotating the real world with knowledge-based graphics on a see-through head-mounted display. In *Proceedings of Graphic Interface '92* (GI '92), 78-85.

[16] Michele Fiorentino, Giuseppe, Monno, Pietro A. REnzulli, Antonio E. Uva. 2003. 3D Sketch Stroke Segmentation and Fitting in Virtual Reality. In *International Conference on the Computer Graphics and Vision*, 188-191.

[17] Mark A. Froehlich, Salman Azhar. 2016. Investigating Virtual Reality Headset Applications in Construction. In *Proceedings* of 52nd Associated Schools of Construction Annual International Conference.

[18] Steffan Gauglitz, Benjamin Nuernberger, Matthew Turk, Tobias Höllerer. 2014. World-stabilized annotations and virtual scene navigation for remote collaboration. In *Proceedings of the 27th Annual ACM Symposium on User Interface Software and Technology* (UIST '14), 449-459.

[19] João Guerreiro, Daniel Medeiros, Daniel Mendes, Maurício Sousa, Joaquim Jorge, Alberto Raposo, Ismael Santos. 2014. Beyond Post-It: Structured Multimedia Annotations for Collaborative VEs. In *Proceedings of the International Conference on Artificial Reality and Telexistence Eurographics Symposium on Virtual Environments* (ICAT-EGVE '14), 55-62.

[20] Thomas Hilfert, Markus König. Low-cost virtual reality environment for engineering and construction. 2016. In *Visualization in Engineering*. Issue 4, Article 2.

[21] Thomas Jung, Mark D. Gross, Ellen Yi-Luen Do. Annotating and sketching on 3D web models. 2002. In *Proceedings of the 7th International Conference on Intelligent User Interfaces* (IUI '02), 95-102.

[22] Reiko Kadobayashi, Julian Lombardi, Mark P. McCahill, Howard Stearns, Katsumi Tanaka, Alan

Kay. 2005. Annotation authoring in collaborative 3D virtual environments. In *Proceedings of the 2005 International Conference on Augmented Tele-existence* (ICAT ‘05), 255-256.

[23] Lan Li, Fei Yu, Dongquan Shi, Jianping Shi, Zongjun Tian, Jiquan Yang, Xingsong Wang, and Qing Jiang. 2017. Application of virtual reality technology in clinical medicine. In *American Journal of Translational Research*, 3867-3880.

[24] Teresa Matos, Rui Nóbrega, Rui Rodrigues, Marisa Pinheiro. Dynamic annotations on an interactive web-based 360º video player. In *Proceedings of the 23rd International ACM Conference on 3D Web Technology* (Web3D ‘18), 1-4.

[25] Benjamin Nurernberger, Kuo-Chin Lien, Tobias Höllerer, Matthew Turk. 2016. Interpreting 2D Gesture Annotations in 3D Augmented Reality. In *Proceedings of 2016 IEEE SYmposium on 3D User Interfaces (3DUI ‘16),* 149-158.

[26] Sebastian Pick, Torsen W. Kuhlen. 2015. A Framework for Developing Flexible Virtual-Reality-centered Annotation Systems. In *Proceedings of 2015 IEEE 8th Workshop on Software Engineering and Architectures for Realtime Interactive Systems* (SEARIS ‘15)**,** 10-17.

[27] Ainsley Sutherland. 2016. Glossy: Audio Annotation in Virtual Reality. In *Buzzfeed: Open Lab*. Retrieved May 11, 2020 from https://www.buzzfeed.com/ainsleysutherland/glossy-audio-annotation-in-virtual-reality.

[28] Po Yeng Tseng, Harald Haraldsson, Serge Belongie. 2019. Annotate All! A Perspective Preserved Asynchronous Annotation Systems for Collaborative Augmented Reality.

[29] Jennifer N.A. Silva, Michael Southworth, Constatine Raptis, Jonathan Silva. 2018. Emerging Applications of Virtual Reality in Cardiovascular Medicine. In *JAAC: basic to Translational Science*. Volume 3, Issue 3.

[30] Haijun Xia, Sebastian Herscher, Ken Perlin, Daniel Wigdor. 2018. Spacetime: Enabling Fluid Individual and Collaborative Editing in Virtual Reality. In *Proceedings of the 31st Annual ACM Symposium on User Interface Software and Technology* (UIST ‘18), 853-866.