IMPROVING DIGITAL OBJECT ALIGNMENT AND REAL-TIME VISUALIZATION VIA REAL-LIFE OBJECT TRACKING

Joseph Santiago Spaeth

Charlottesville, Virginia

Bachelor of Science, University of Virginia, 2021

A Thesis submitted to the Graduate Faculty
of the University of Virginia in Candidacy for the Degree of
Master of Science

Department of Computer Science

University of Virginia

May 2023

Tariq Iqbal, Chair

Afsaneh Doryab

Seongkook Heo, Advisor

Improving Digital Object Alignment and Real-Time Visualization via Real-Life Object Tracking

Joseph Santiago Spaeth

(ABSTRACT)

Augmented reality provides a unique opportunity to interact with both physical and digital objects simultaneously. However, the process of visualizing digital objects alongside physical ones can cause friction, particularly in cases where the user desires precise alignment of the digital and physical objects. To resolve this we offload the process of moving digital objects with one's hand to the use of a marker instead. By having an independent object handle the position and rotation of a digital object one can place the marker in an arbitrary place and ensure it is both static and non-rotating, allowing for more precise interactions regarding placing and inspecting the object in question. Object tracking inherently handles position and orientation in 3D space, and we independently handle scaling using a context-sensitive menu disconnected from the object. This also allows for more accurate visualization of digital objects alongside large physical ones which may not be easily manipulated by a user. We compare this implementation to a baseline system based on the Microsoft Mixed Reality Toolkit's (MRTK) implementation of these object manipulation interactions to determine its efficacy and usability compared to existing methods. After testing our object tracking implementation against a baseline implementation of object transformation, rotation, and scaling. This implementation relies on mid-air gestures, where one pinches and grasps an object to translate and rotate it, and uses a 2 handed pinch-and-stretch input for scaling, as well as a 3 DOF scaling option where one can

pinch and drag the corners of a bounding box surrounding the object to scale it along each axis independently. We find that there is a statistically significant difference in completion for 3 of the 4 tasks we asked users to perform and a significant difference in their preference toward our system over the control. This implies object tracking with markers is both a more efficient method of aligning and scaling objects as well as better accepted by users.

Dedication

This is for my cat, Tamale, who will never understand just how important she is.

Acknowledgments

This would not have been possible without the help of Dr. Heo, he has been with me every step of the way through the process of completing this paper. I am immensely grateful for him, the classes he has taught, and the interest in Human-Computer Interaction he helped to stoke in me. I would also like to recognize the support of both my family and friends. This program has been immensely challenging, as has life been in the last two years. I would not be here without the people who let me lean upon them. Finally, I would like to recognize the participants in our human-subjects study, as without them there would be no paper to present and I greatly appreciate the time they took to help complete this study.

Contents

List of Figures

Chapter 1

Introduction

Several augmented reality applications exist to allow users to manipulate 3D digital objects in conjunction with the existing physical environment. Be it a simple overlay such as in a game like Pokemon Go [15], or something as complex as interactive educational and vocational training tools [6, 10, 3]. However, as AR apps become more common and rely on more complex interactions, it becomes more important to be able to finely manipulate 3D objects and ensure they are correctly placed in the environment and/or correctly placed relative to important physical objects within the user's field of view (FOV).

Typically, both in and outside of AR/MR applications it is a common solution to reduce the degrees of freedom for an object's transformation in order to more easily perform fine-grained manipulations of the digital object [2]. For example, in many 3D modeling programs [17] or game engines [16], a digital object can be translated and rotated along any of its cardinal axes. This can also be handled by snapping objects to surfaces, an approach used by SnapToReality [13] and Snap-to-fit [7]. Though limiting degrees of freedom is useful when handling precise manipulations, this creates a clear trade-off with efficiency in these interactions. This demonstrates clear drawbacks with both 6 DOF (3 translational and 3 rotational degrees of freedom) manipulation and limiting DOF for more precise movement. Both of these paradigms can make it difficult for the user to align digital and real-life objects, so we propose using props as a means of better aligning digital objects with the physical world, as one can simply interface with two physical objects without having to worry about issues with imprecise movement or limiting one's ability to move and rotate an object as they interact with it. Particularly given that it is a challenge to align objects when interacting with them using hand tracking, we believe tracking props will be a more elegant solution to the problem.

Several other systems already integrate props, either in the form of specialized controllers [4, 27, 14], or tangible UI projects relying on arbitrary props [11, 5, 9]. In the case of the Wii, PSVR, and Oculus Rift, all present input systems rely on tracking a controller(s) held in one's hands. This provides a pointer on the Wii and a metaphor for one's hands in the VR systems. The tangible UI projects provide means of using arbitrary props in one's environment to help interact with the computer in some way, for example, in MixFab they can use arbitrary objects as props to assist in modeling objects. They use markers to create negative wells in a desk holder for pens. All of the papers mentioned there rely on a significant amount of additional hardware in order to make these interactions work, making them untenable for practical use. Aside from a printer and HMD however, no other specialized hardware is needed for our solution. By freely manipulating physical objects, the user can easily finely manipulate the digital one bound to it and does not need to rely on potentially expensive hardware solutions.

In order to improve precise object alignment and scaling tasks, we propose a solution of the usage of tracked markers to allow the user to finely manipulate the position and rotation of a digital object affixed to it. We separate scaling completely as an interaction in order to ease the cognitive load on the user and ensure the ability to perform precise scaling. We claim our method allows for more precise object alignment, simpler manipulation of the digital objects as users already understand how to manipulate the physical markers, and that it benefits from not requiring specialized hardware, unlike other potential solutions. This will allow for better object visualization for makers, as they can verify their prints will come out as desired prior to printing. It can also be useful as a means of placing AR content in one's environment, as one can put down a marker and leave it resting in the latent environment, freeing the user's hands for other interactions, similar to SnapToReality[13]. To test this control method we compare it to a control implementation of object manipulation controls based on out-of-the-box UI/UX options from the Microsoft Mixed Reality Toolkit [25].

Chapter 2

Related Work

Here we discuss several works relevant to the production of this one. This includes the topics of object manipulation and visualization, DOF restrictions, and the pieces of software leveraged to produce the software used for our study.

2.1 AR Object Manipulation and Alignment

There exist several issues with manipulating objects in Mixed Reality, the biggest being issues with general object alignment, separation of degrees of freedom in manipulation, visual gaps between the digital content and objects in the real world, and imprecision in object manipulation [2, 6]. Additionally, AR apps are often created for substantially different target platforms, with some relying on the use of a cellular device or a tablet with a touch screen, and others using a Head-Mounted Display (HMD). In the latter case, the touch screen affords additional interaction methods not afforded to the HMD user which must be considered.

2.1.1 Object Alignment and Visualization

A common issue in AR/VR applications is understanding the depth of objects and their size, particularly in the case of AR, understanding the size of digital objects relative to physical counterparts and where digital objects lie relative to the physical environment surrounding them. Solutions to this have been explored, such as Barerra-Machuca et al's Context Based 3D grids [21], which allow a user to pull up a context-sensitive gridding system better allowing them to understand the size of both extant physical objects, and the digital ones they wish to manipulate via gestural control. Object snapping is another solution to this issue, which is discussed in further depth in the following subsection.

Some object visualization solutions have been proposed in the form of specialized modeling software, but these require substantial amounts of extra hardware support to use or are not designed with AR in mind. Solutions such as MixFab [11] and Reprise[12] both allow for the visualization of physical and digital objects simultaneously, but require extra tools and domain knowledge to properly use. Another helpful control metaphor is Englmeier et al's Spherical Proxy controller [22], which is a beneficial way to handle 6 DOF movement, but also suffers from requiring specialized hardware.

The use of Physical Markers is also a relatively explored area, but less so for the explicit use case of object alignment. It has been explored in AR apps to help visualize 3D models, but this was in conjunction with webcams and a computer instead of an HMD [8]. They have also been used in other applications but for static displays and/or supporting other target devices [18]. It is also possible now to examine a target and obtain pose and size data on it with marginal error in a reasonable amount of time [20], however, the authors do not apply this for alignment tasks, merely visualization. Broadly, the usage of arbitrary props has been explored for a significant amount of time, with the Tangible Bits being an exploratory paper establishing methods of creating and using Tangible UI, it is not a work explicitly operating in the AR space however [1]. Work has also gone into offloading the process of recognizing given targets to ease the load on the device running the AR software.

2.1.2 Restricting Degrees of Freedom

The use of constraints is a typical solution to 3D dimensional object alignment. For example, object snapping is a typical method presented by multiple authors, such as Neurnberger et al with SnapToReality and Noghabei and Han's snap-to-fit function [13, 24]. Object snapping is a reasonable solution to the issue of alignment, but these cases are not particularly flexible. A function like snap-to-fit allows for the simple pairing of digital objects but only works because a ground truth alignment is already known. SnapToReality fairs better in this regard as it finds surfaces to snap to in real-time. However, it fails in handling dynamic constraints (the authors left this to further work), which a system relying on physical markers better resolves (as long as the tracked object remains in the FOV, any constraints applied to it will also apply). The work also relies on using relatively large surfaces, and the objects still need to be aligned on those, SnapToReality merely ensures things such as preventing an object from being placed within or above an extracted surface instead of atop it.

Another way to resolve this aside from explicitly limiting DOF is by adjusting how fine-grained one's movement is. This would include approaches such as adjusting C-D ratios [2], or using gestural control to restrict DOF in real-time [19]. In both cases, contextual cues are used to limit the inputs of the user. By adjusting the control-display ratio of object movement, rotation, or scaling, one can make larger physical movements correspond to smaller movements of the digital object, making it simpler to achieve the desired outcome for the user. Similarly, by making DOF restrictions contextual as Plane Ray Point does, one can make large, imprecise movements and then more easily make fine adjustments to achieve an ideal alignment after the fact[19].

Chapter 3

Designing Precise AR Object Manipulation

Two interaction methods were designed for the study, the first of which is a marker-based method leveraging Vuforia's Image Target detection capabilities, and the second is a baseline implementation based on the default object manipulation tools provided by Microsoft's Mixed Reality Toolkit (MRTK) [26, 25]. In addition to designing means of controlling objects' positions and orientations, scaling methods were also created for the two interaction methods. A menu allowing for scaling along each axis was implemented for the marker-based method, and two different types of scaling are used for the MRTK implementation. Microsoft's default pinch-and-stretch interaction is supported, as well as an ”independent scaling mode” which toggles on and off a bounding box that a user can use to scale an object in three dimensions independently of each other.

3.1 Interaction Design

3.1.1 Physical Markers

Tracking physical markers inherently solves the issue of keeping track of both a digital object's position and rotation in 3D space, as one merely needs to determine the position and rotation of the prop being used to track the object. Though this is similar to manually manipulating the digital object, picking up a physical prop has several benefits to the user. Users have a firmer grasp on the placement of physical objects than digital ones, as it is easier to keep track of the relative depth and size of the physical object relative to other extant objects [2]. Though the camera in the headset still must determine the position and orientation of the object, placing and orienting it in 3D space becomes a simpler task as the user can more readily intuit the results of manipulating the physical prop. It is also a simpler task to keep these objects still, for example, if one wished to rotate a digital object on a table, merely spinning a marker is easier and allows for finer control of one's movements than merely tracking one's hand. This is both due to the micromovements involved in manipulating an object which can create undesired results (as users may not realize their hand is moving) and the fact that the points being tracked by the headset do not move relative to each other. This makes determining the position and orientation of the marker as less intensive task compared to hand-tracking.

3.1.2 Object Snapping

An object-snapping implementation was created as a part of tracking the markers. The object-tracking implementation relies on the digital object maintaining the same positional coordinates as the tracked object within Unity, if the pivot point for the digital object is not aligned with the pivot point of the Image Target, then scaling operations can cause the pivot to move relative to the target and potentially move the entire digital object relative to the target.

We resolve this problem with a script that ”snaps” the object into place upon any scaling change. It operates under the assumption that the pivot needs to be moved along a cardinal direction (though this can be generalized) in the object’s local coordinate system. The desired direction on the X, Y, or Z axis is selected, as is whether or not motion toward the marker occurs in the positive or negative direction along the selected axis. After any scaling operation, a ray is cast in the selected direction. The ray extends outward until it collides with the game object representing the marker, and then the object is translated along the ray until it collides with the marker. This can ensure the object remains in place while still allowing for changes in any direction. Additionally, as this script is called during any update when the object is being scaled, the translation occurs as the scaling operation occurs. Makes it so the scale of the object can be updated in real-time while maintaining the proper position and orientation of the manipulated digital object.

3.1.3 Scaling

Though the detection of physical markers is a means of handling the position and orientation of a digital object, there is no natural extension of this control metaphor to scaling the digital object. As such, it is offloaded to a separate menu, where multiple scaling options are implemented. A context-sensitive menu can be displayed by raising one’s left hand which displays several options for scaling the object tracked by the

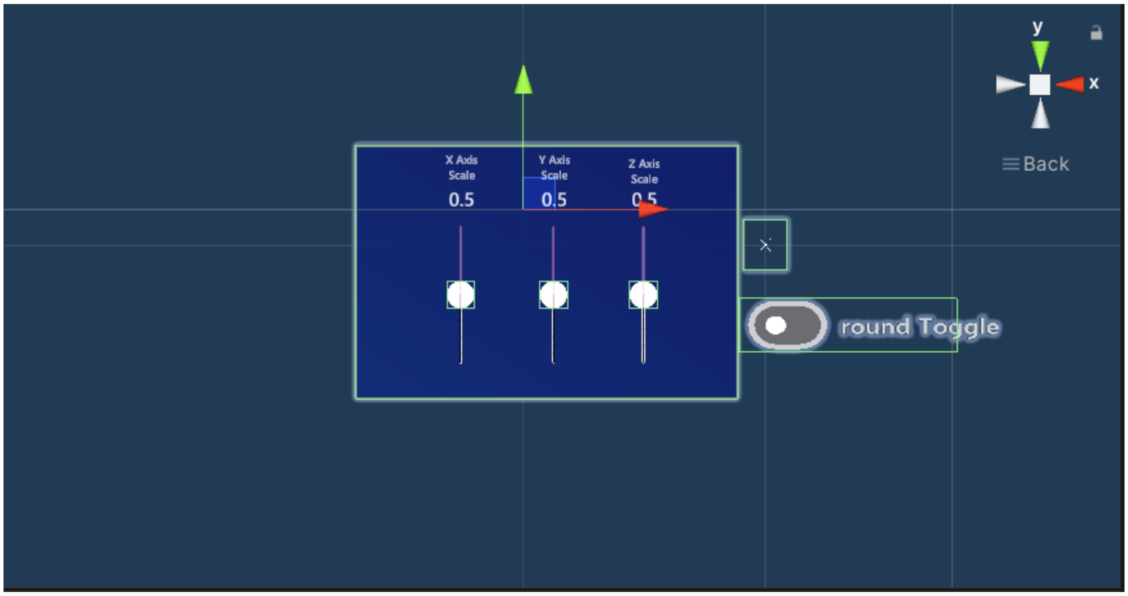

Figure 3.1: The context-sensitive menu used to scale objects when they are displayed through Vuforia

marker. We choose to make this menu context-sensitive so that it does not overwhelm the user. We elected to provide them with sliders allowing them to manually adjust the scale in each dimension. This allowed for simple independent scaling in each dimension, as well as placing constraints on the possible size of the object to ensure they remain easy to work with and visualize.

3.1.4 Baseline Methods

The standard methods of control implemented are based upon default controls provided by the MRTK, with some constraints placed upon them to be made more similar to the input methods we provide with the object-detection portion of our testing software. In this case, 3D objects are able to be grasped using a one-handed pinching motion and moved as long as the user continues to ”hold” the object. Scaling is handled with a context-sensitive bounding box around that object which can

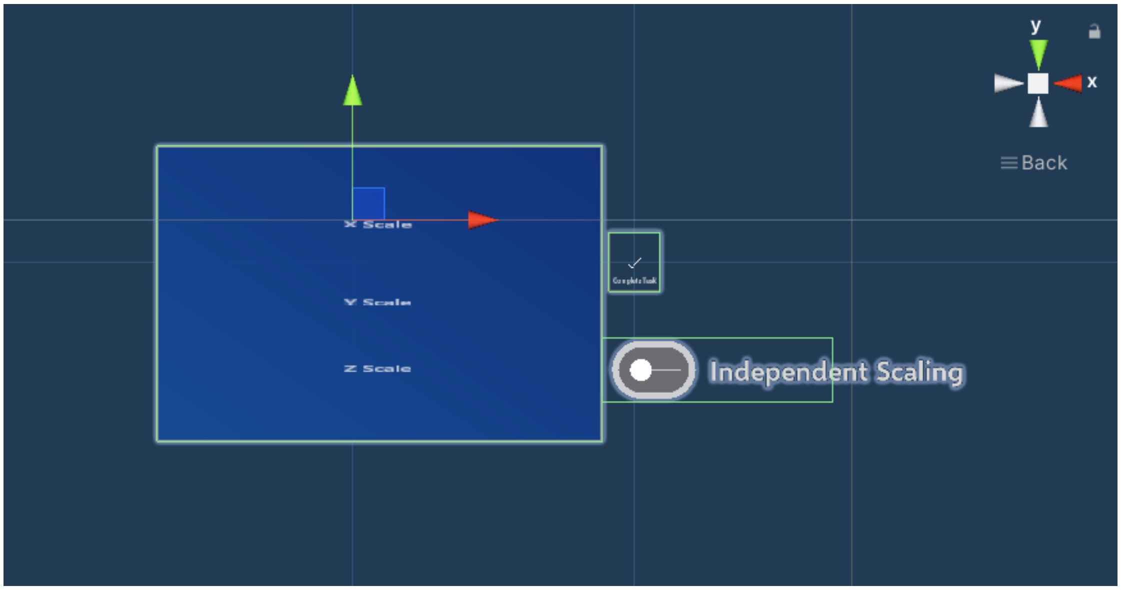

Figure 3.2: The context-sensitive menu used to display object scale and enable independent scaling when manually manipulating digital objects

be used to scale or rotate the object along any of its cardinal axes, and a standard two-handed pinch-and-stretch input to scale an object equally in all three dimensions. The bounding box can be activated by pressing a toggle on a context-sensitive menu. We elect to do this to minimize extraneous inputs on the display, attempting to avoid stressing the user. The menu also displays the current scale of the object one is scaling which can be seen in Figure 3.2.

This input method is meant to exist as an intuitive baseline to act as a control for the study. The control metaphor is simple, as it extends to how one would naturally interact with these given objects (by grasping and holding). The only unnatural movement is scaling, which lacks an obvious real-life parallel. However, the two metaphors used for scaling both have common analogs in other digital applications, pinch and stretch scaling is analogous to zooming on a touch screen device, and the bounding box implementation is alike to resizing a selection in image-processing software. As such the control implementation is designed in a manner such that users

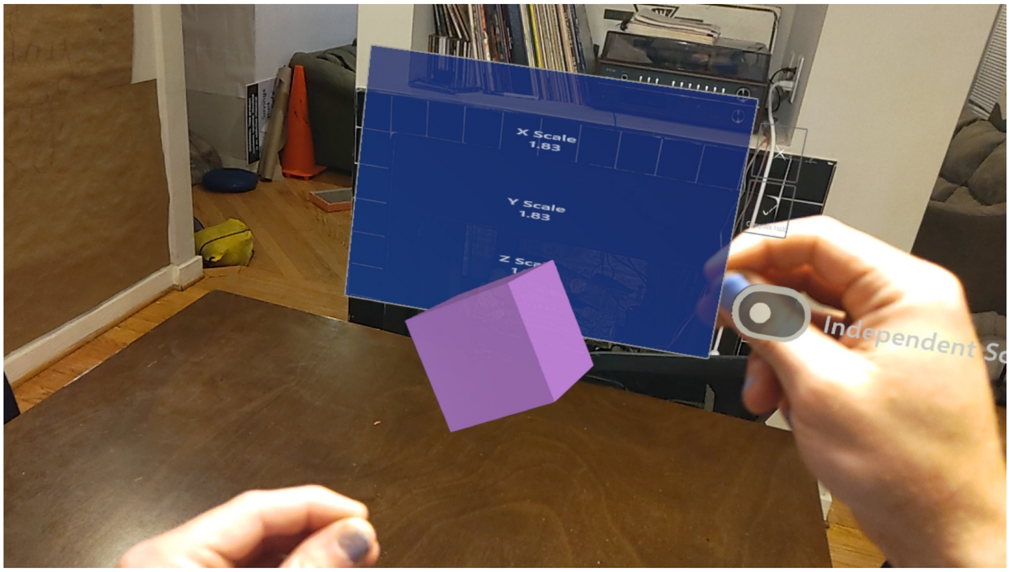

Figure 3.3: An example of two-handed pinch-and-stretch scaling

can pick it up with minimal excise tasks involved in the manipulation of a given digital object(s).

3.1.5 Scaling using Manual Manipulation

As previously stated, two methods of scaling are used in our design of baseline object manipulation. The first scaling method is a two-handed pinch-and-stretch method, as seen in Figure 3.3. The user grasps the object with both hands and then brings them closer together or further away from each other in order to make the object smaller or larger respectively. This scales the object equally along all three dimensions. The second implementation is slightly more complex and relies on the MRTK's bounds controller bounding box implementation. When the user triggers a toggle, a bounding box appears around the object they wish to scale. They can grasp the corners and move them freely in all directions. The object is then scaled accordingly to match the bounds of the bounding box, which are being manipulated by the user. Each

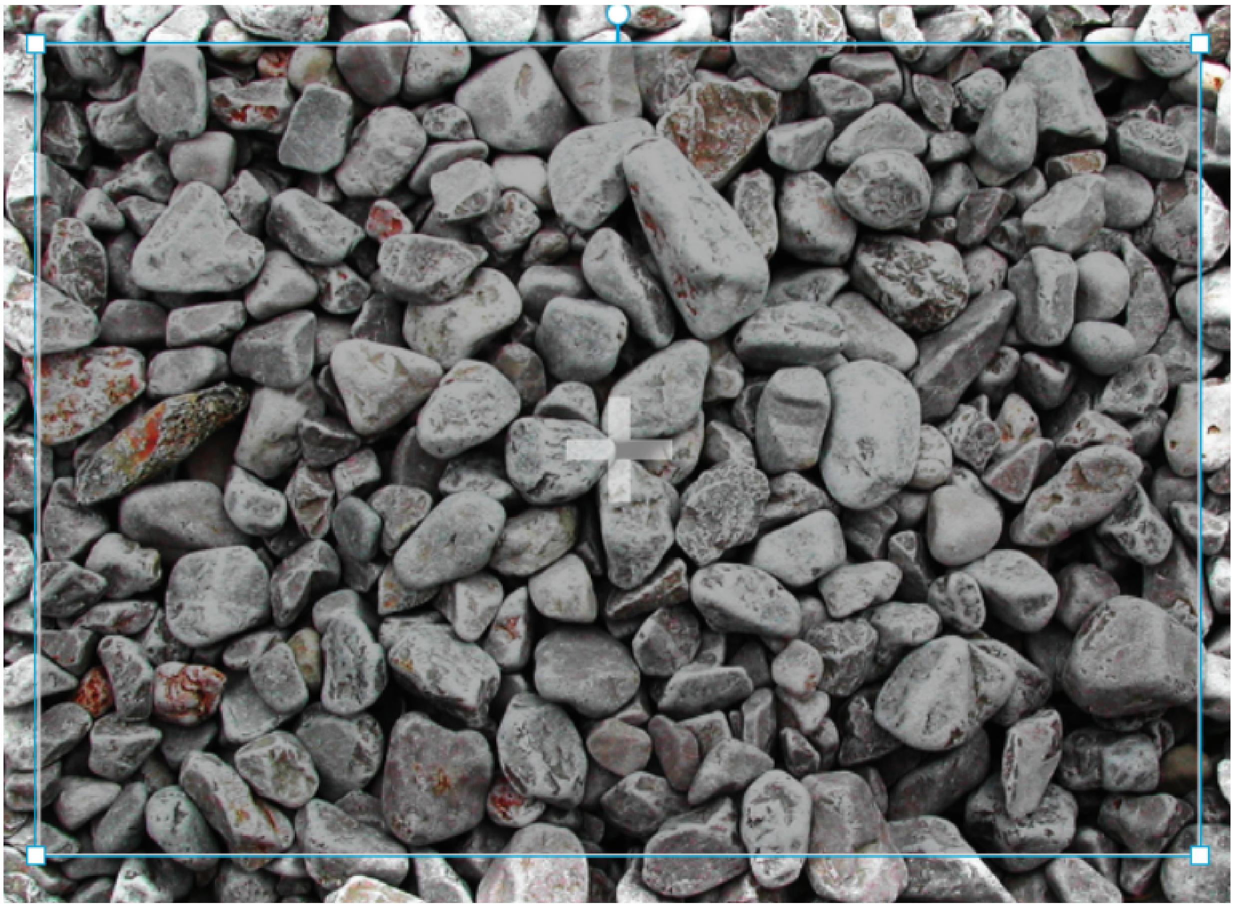

Figure 3.4: A 2-D image scaler from the image processing software Pixlr

dimension scales separately, allowing for 3 DOF scaling.

There were two motivating decisions to design the scaling interaction in this manner. The first is its similarity to the first method of scaling in all three dimensions. In both cases, the interaction is predicated upon pinching and stretching the digital object in question. This makes switching between the two methods bear less of a cognitive load on the user. Because of their similarity, it is less difficult for the user to adapt to the second, more complex method of scaling when the task(s) they are provided with call for its usage. As previously discussed as well, this bounding box method is analogous to scaling methods in 2D image processing software. This in conjunction with the similarity to the first interaction helps to greatly reduce excise tasks for the user, as they can intuit how to scale the object from the bounding box and apply knowledge from the new experience of scaling 3D objects from the uniform scaling interaction.

3.2 Interaction Implementation

Here we describe some of the technical details involved in how the interactions described in the prior section were created. We also discuss two of the toolkits leveraged to create our testing software.

3.2.1 Physical Markers

To create the markers tracked by our software, premade markers from Vuforia were printed, and we also used Gasques, Scheurer, and Lopes' AR Marker generation tool [26, 23]. The tool allows for the creation of random targets of different sizes and feature (Vuforia's term for points of interest tracked by its tracker) counts. One of these can be seen in figure X. Though the markers can be printed on anything, they were provided to the users on entire 8.5" x 11" sheets of paper, affixed to a prop for the user to use, or they were given a smaller marker reinforced with card stock. In all cases, the markers were easily manipulated, as they can be tracked once the camera is able to focus on them once.

The markers are set up to render 3D content over them when detected by the Hololens. Within Unity, the digital objects are set up to render when an image target is detected and tracked, and rendering is turned off when the object is lost (this operates under

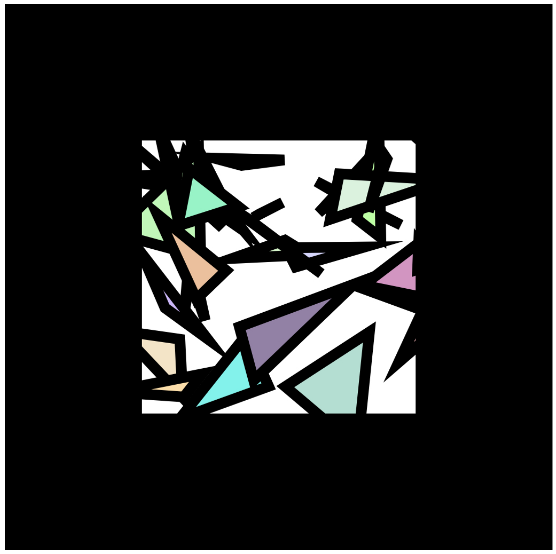

Figure 3.5: A sample marker generated by the AR Marker tool.

the assumption that this means the object is no longer in the user's FOV and thus does not need to be rendered). The user cannot directly interact with the object, any changes to its position and rotation must be done by directly interacting with the markers they are associated with. It is also worth noting that other behaviors can be implemented upon marker detection or loss instead of turning the object's renderer on and off. It can be made to remain in place, be made into a manually manipulable object, or anything else, as one can call an arbitrary number of methods upon the beginning or end of the tracking process.

3.2.2 Scaling using Physical Markers

We use a separate context-sensitive menu to handle scaling when tracking objects. The menu, as displayed in Figure 3.1, contains a set of sliders and a toggle. The sliders each independently control the scale of the object in their labeled dimension (local X, Y, and Z axes). The sliders allow the user to scale the object as high as

3 times its base size to as little as $\frac{1}{3}$ of its base size. The toggle to the right of the menu enables a rounding mode. The base implementation of these sliders allows the user to pinch them to adjust the value of the slider. However, this still allows for problems with capturing fine-grained motion and user intentionality, as it is difficult to precisely manipulate the slider to the desired value. When rounding is enabled, the values of the slider near key commonly desired scales $(.5, 1, 1.5, 2, \text{and } 2.5)$ all map to those respective numbers. The value is controlled by the developer but in our implementation maps values within .05 above or below those thresholds to the same threshold value. This makes it easier to capture the user's desired behavior while still maintaining the desired interaction method. A more elegant solution is likely manipulating the C-D ratio when the slider's value is around one of those key thresholds, making it more difficult to move past one of those thresholds and easier to move up to one of them. We leave that alternative scaling implementation to future work.

3.2.3 Baseline Object Manipulation Implementation

Little needed to be done in order to implement the manipulation and rotation controls we wished to support. The MRTK comes with out-of-the-box support for them, meaning all that needed to be done was implement appropriate constraints such that the user could not accidentally manipulate the objects from a far distance, or resize them to inappropriate sizes.

3.2.4 Baseline Scaling Implementations

The baseline control method uses two control schemes for scaling, the first is a pinch-and-stretch method which comes from the MRTK without any further adjustment needed, aside from placing constraints on the objects to stop them from becoming too small or too large. Little needed to be done in order to correctly implement this. The second scheme relies on the MRTK's bounding box controls and is a more complex control method to implement. It allows one to grasp the corners of the bounding box and then manipulate its scale by moving one's hand while grasping those corners.

The second scaling method is implemented with a bounding box around the object which can be toggled on and off. The bounding box is bound to a box collider that is as large as the object at its longest point in each dimension. The bounding box has points attached to each of its corners (represented by grey cubes) which can be grasped by the user. Once they grasp the point it can be moved and the object scales in all three dimensions according to the movement of the user's hand while it remains closed. Motion left and right relative to the user will scale the object along the X axis, up and down along the Y axis, and forward and backward along the Z axis. Unlike in the prior implementation, however, these movements are tracked independently, allowing for the scaling to be separated along each axis.

3.2.5 Software Used

This work relies heavily on two pieces of preexisting software, Vuforia, an AR development toolkit, and Microsoft's Mixed Reality Toolkit (MRTK). Here we discuss those toolkits used and their applications, specifically focusing on the portions of them

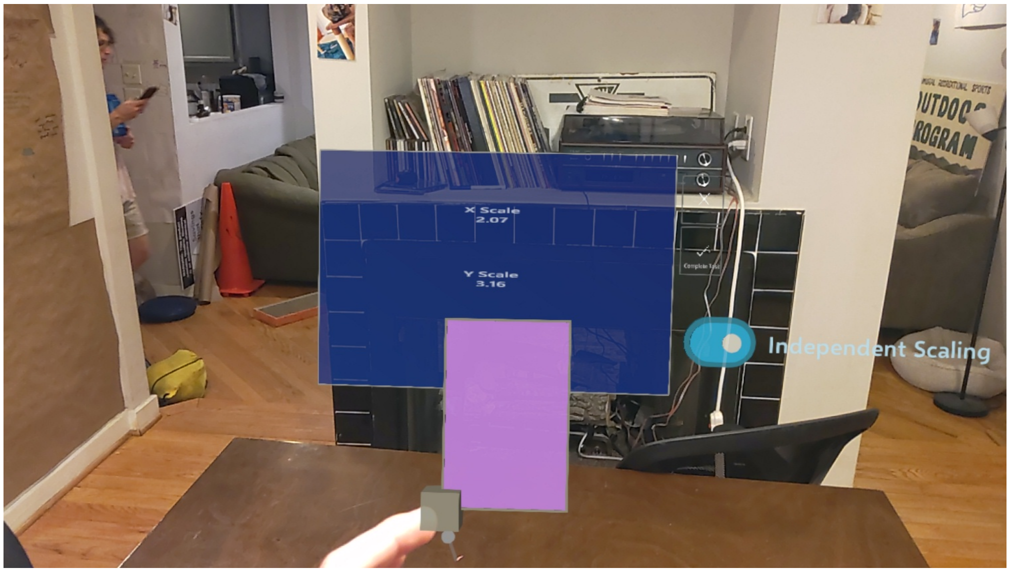

Figure 3.6: Use of the independent scaling mode.

leveraged for this project. The software created was developed within the Unity Engine, and the toolkits exist as UnityPackages which were included within the project [16].

Vuforia

This project relies heavily on the Vuforia engine. It is a popular platform for AR development and supports both HMDs and mobile devices as target platforms. One of the primary reasons for its popularity is its integration of target detection. The engine is capable of detecting several different types of targets (images or collections of images arranged on a cylinder or box), Object Targets (three-dimensional objects), and ”Area Targets” whole recognizing and augmenting entire environments based on 3D scans of the environment. This work leverages Vuforia’s ImageTarget and MultiTarget detection systems to recognize the presence of markers and a box within the FOV of the Hololens in order to display content to the user[26].

MRTK

The Microsoft Mixed Reality Toolkit (MRTK) is a project containing assets and components for mixed reality development. It acts as a framework within Unity (as well as the Unreal Engine) upon which AR apps can be built, and elements can be used as is, iterated on, or replaced entirely while handling the core aspects of mixed reality interaction for the developer. It is the backbone of many mixed reality applications, particularly those targeting the Hololens. This project relies heavily on its UI assets as well as its implementations of digital object grasping, rotation, and scaling [25].

Chapter 4

Evaluation

To measure the effectiveness of object tracking in comparison to the standard AR method of manual place meant a user study was conducted. The study involves the completion of multiple object alignment and scaling tasks both using and not using the object tracking implementation. We collect data on completion time and quality of alignment (note that the data on the quality is qualitatively measured). Additionally, participants were asked to complete a questionnaire about the user experience after using the software.

4.1 User Study Design

A 2x4 within-subjects study consisting of 4 tasks split between 2 conditions (for a total of 8 tasks) was conducted. The conditions for the study were whether Physical Marker tracking was enabled, we refer to the condition with object tracking enabled as Object Tracking with Markers, and without it enabled as Manual Object Placement. Tasks were:

- Setting a digital cube on a flat surface in front of the user and scaling it in all dimensions. This was a simple task designed to help users get oriented with the system and establish comfort with the controls.

- Aligning a digital ”pop-socket” accessory with the center of a physical cell phone case and scaling the digital object in 2 dimensions. This tests the user’s ability to align the objects, as well as handle scaling in separate dimensions. It forces the user to use independent scaling mode.

- Adding a digital roof to a physical box and scaling it in 2 dimensions, such that one created an overhung roof on the box. This task also forced the use of independent scaling as well as ensured the user took time to verify the alignment of the object. This forced a more detailed visualization of the digital content.

- Placing 2 digital objects on a flat surface in front of the user, then scaling and aligning those two objects such that they formed a bridge between an artificial gap in the desk in front of the participant. This task forced the user to engage with both physical and digital object alignment, as well as use the scaling tools. It also demonstrates a use case where one may wish to visualize a specific set of objects but certain physical objects in the environment may be immovable (in this case the objects creating the gap).

Each task was completed both with and without object tracking enabled. The users were asked to complete each task twice, with minor alterations to each task when performing them for the second time, with an exception being made for the final task in the study. These alterations were scaling the object down instead of up in Task 1, replacing the phone case with a smaller one for Task 2, and making the roof align with the edges of the box instead of hanging over them in Task 3. When object tracking was disabled the digital objects needed to be placed manually via a one-handed grab input, and scaled using a two-handed grab input or by pressing a toggle allowing one to scale the object independently in each dimension. Additionally, though presented as such,

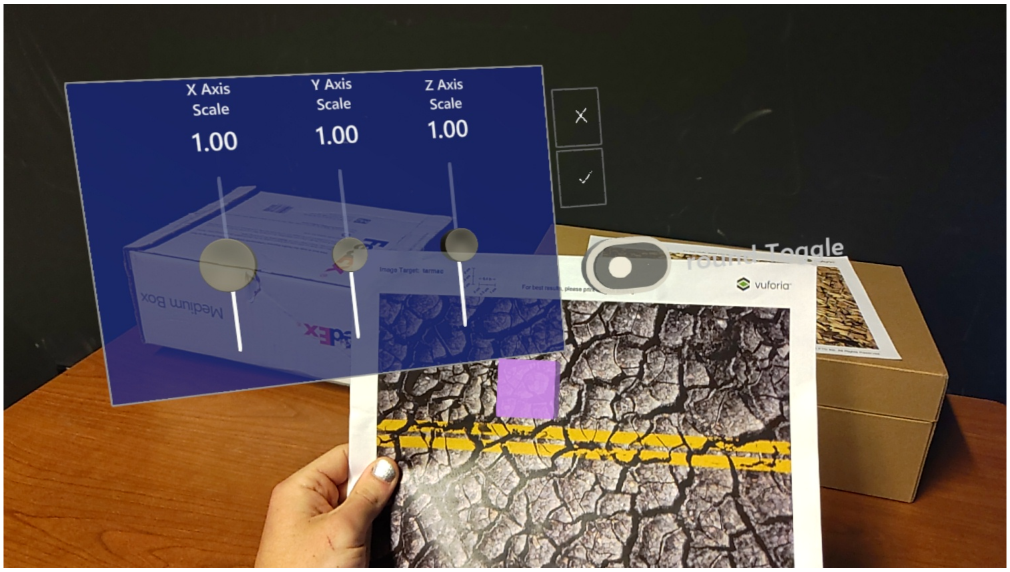

Figure 4.1: Screenshot of Task 1.

the users were allowed to perform their actions in whatever order they preferred. In the case of manual manipulation tasks, it is typically simpler for the user to manipulate and scale the object(s) prior to placing them on a surface, instead of placing them and then ensuring the scale object(s)' scale is correct. Figures 4.1-4.4 show the initial conditions of the physical marker tracking versions of various tasks.

The study was counterbalanced in order to reduce learning effects between conditions. Half of the participants were asked to perform the object-tracking version of each task prior to completing the manual version, and the other half of the participants were asked to perform the manual tasks first. After the participant has completed their tasks they are given the post-study questionnaire. The questionnaire consists of a mix of 5-point Likert Scale questions as well as a small set of open-ended questions for the participants.

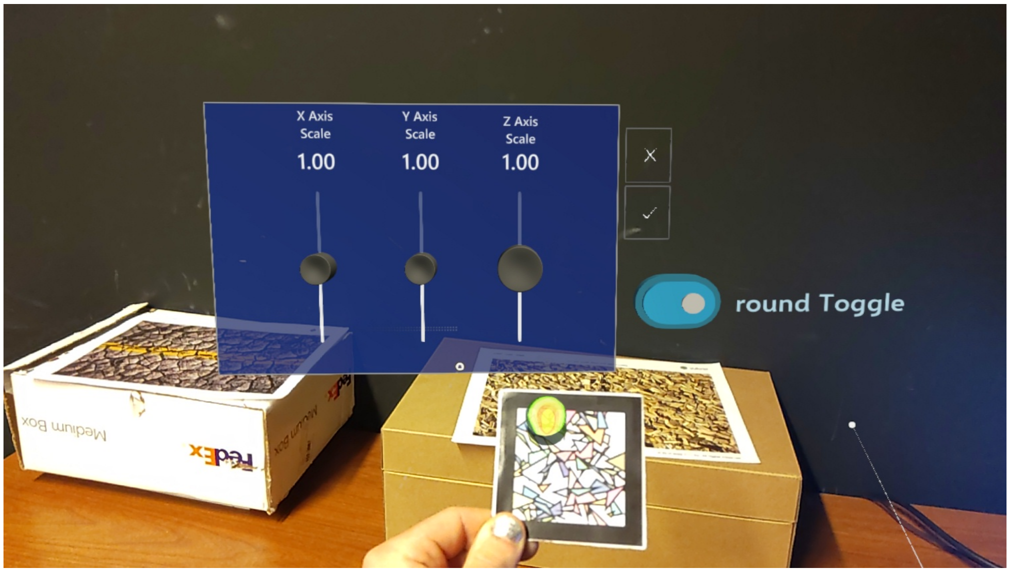

Figure 4.2: Screenshot of Task 2.

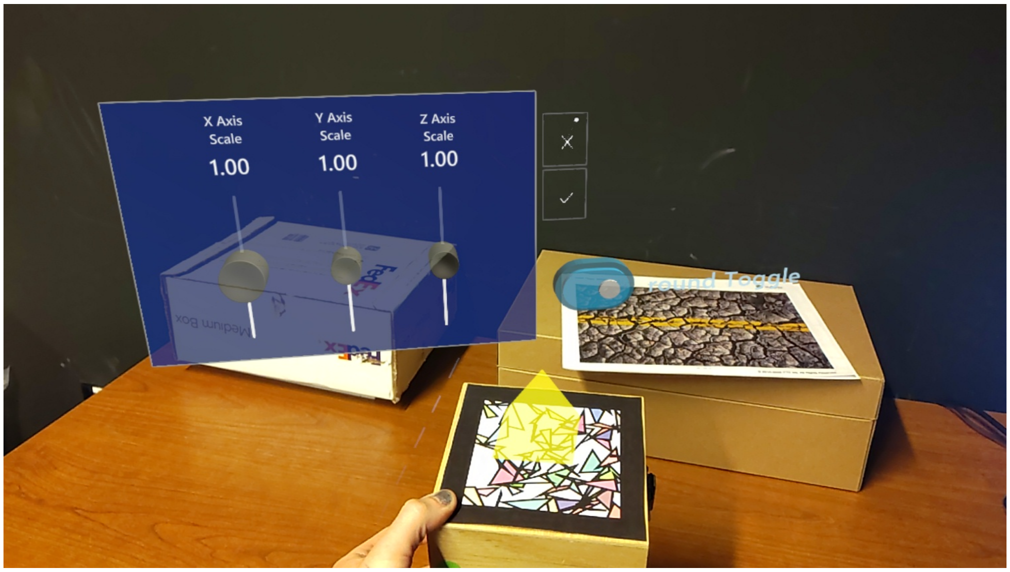

Figure 4.3: Screenshot of Task 3.

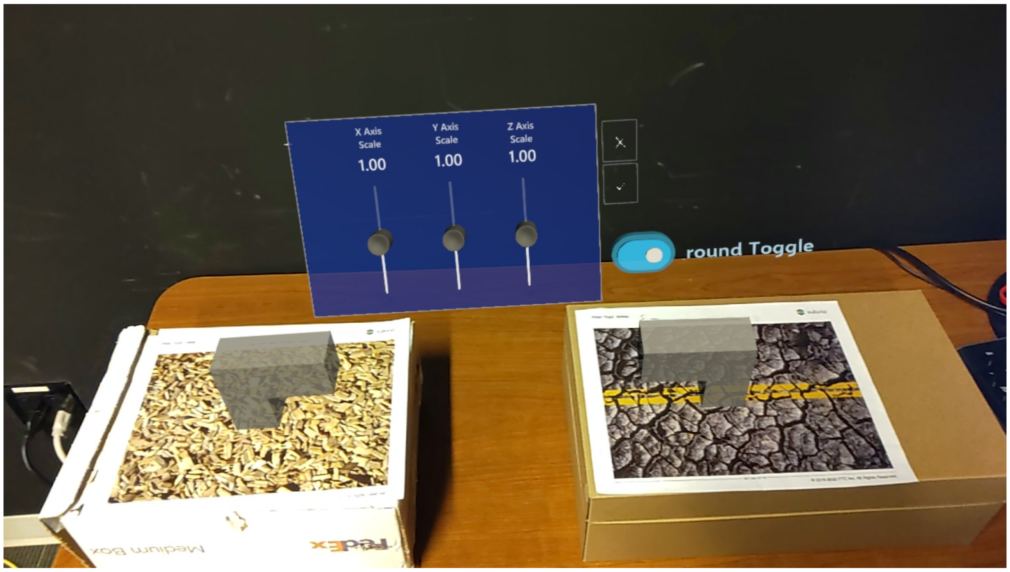

Figure 4.4: Screenshot of Task 4.

4.2 Participants

The user study had 12 people, containing 8 men, 4 women, and no non-binary and/or gender non-conforming people. Their age range was 18-32 (with a mean age of 23.83). Additionally, they indicated varying levels of Virtual Reality or Augmented Reality (VR and AR respectively) experience. Six indicated no experience, 5 indicated some experience (< 1 year of experience), and 1 indicated having substantive experience (1-3 years of experience). They also indicated varying levels of 3D modeling experience, with 4 indicating no experience, 7 having indicated some experience (< 1 year of experience), and 1 indicating substantive experience (1-3 years of experience).

4.2.1 Quantitative Measures

We recorded task completion time. This is measured from the beginning of each task to when the user presses a button allowing them to move on to the next task. At that point, a timer running in the background is stopped and the time taken in the task is recorded. This datum is recorded for all conditions. It is not a direct measure of ease of use but is a helpful indicator that one alignment method may be more useful/intuitive than the other. Additionally, it may indicate that although a given method may be more difficult to grasp for a user, the gap in time to achieve desired results may indicate that once a method is learned by the user it is superior due to spending less time completing a given task, allowing for more productive use of AR visualization software.

4.2.2 Qualitative Measures

Several qualitative measurements were taken via a post-study questionnaire. The questions are separated into three categories, alignment satisfaction, scaling satisfaction, and open-ended feedback. These questions serve to provide information on the perceived quality of object alignment and scaling, capture user preferences, and determine which (if any) methods are ideal for 3D object visualization in AR.

To determine how a participant felt about the quality of alignment and scaling they were asked 4 questions. The participants were asked if they could achieve satisfactory alignment or scaling if it was easy to achieve that, if it was efficient to achieve that, and which of the two methods they preferred. Aside from the final question, all of those questions were asked using a 5-point Likert scale. The final question was a forced-choice binary question where the participants must choose their preference between one method or the other. They are then asked to elaborate on why they made the choice they did.

The final question in the survey is an open-ended question asking the participants for feedback on all aspects of the software used in the study. The goal of this is to capture any concerns or specific changes the participants would make to the software. This question allows us to identify avenues for further study as well as to address potential issues with the software as it currently exists and fix it as needed.

Chapter 5

Results

Here we present the results of our user study. We find that both quantitatively and qualitatively object tracking with markers is the superior method for handling object alignment and scaling.

5.1 Quantitative Results

Task completion was measured as a means of measuring the benefits of one control method over the other. On average, object tracking with markers outperformed manual alignment and scaling for all tasks. However, this result was only statistically significant for 3 of the 4 tasks ($p > .05$, $p < .0001$, $p < .05$, and $p < .05$ respectively), Significance was calculated using a two-tailed T-Test. The mean task completion times were as follows:

- Object Tracking with Markers: 78.711s, Manual Object Placement: 83.333s
- Object Tracking with Markers: 62.173s, Manual Object Placement: 110.195s
- Object Tracking with Markers: 61.862s, Manual Object Placement: 96.032s
- Object Tracking with Markers: 84.103s, Manual Object Placement: 117.653s

They can also be seen as a graph in Figure 5.1.

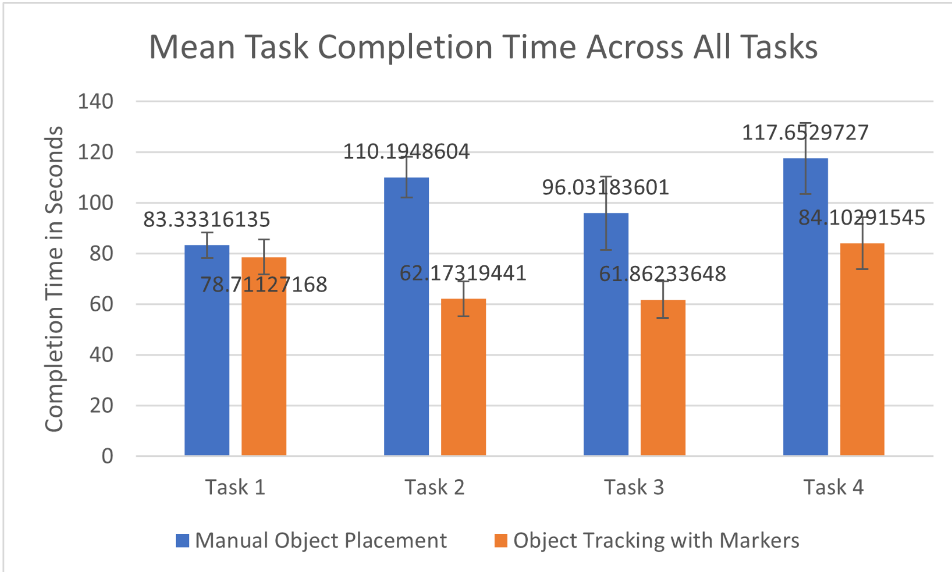

Figure 5.1: Mean completion times across all tasks for both the object tracking with markers and manual object placement conditions.

5.2 Qualitative Results

There were several relevant results from the post-study survey. In all cases, it was found that users had more agreement with the provided Likert-Scale statements on object tracking than manual object alignment. Additionally, in the forced choice questions, users indicated they preferred the object tracking methods.

5.2.1 Alignment

Overall, object alignment using physical markers was much preferred by the study participants when asked the forced-choice question, with 8 indicating they preferred the markers and 4 indicating they preferred using manual object placement to align the objects. One user specifically indicated it was easier to make fine adjustments to the objects when using the markers, saying ”I felt like I was really struggling with perfecting angles and making items flush without the markers, so having them made the assignment way easier on me." Another user said that placing physical objects was simpler, as it was the same act they were already used to doing in real life. When compared to manipulating digital objects they said it made alignment harder as they had less of a grasp on what they were doing. It was also indicated that the manual object placement approach was more "stressful and cognitively loaded", and could even cause cyber-sickness.

One who liked manual alignment said they preferred it as it felt like a more intuitive use of their hands than moving the markers. Another felt it was less translatable to real life than merely moving objects as if they actually existed. They also indicated they did not enjoy context-switching between looking at the physical marker and looking at the scaling menu when manipulating the objects. They felt the rotation and translation made more sense when the digital object was within their grasp. The others did not respond with information relevant to object alignment (one neglected to respond and the other wrote about object scaling instead).

Likert-Scale Responses

Specifically, 8 participants indicated they either somewhat agreed or strongly agreed with the statement "I could align the object with satisfactory quality using Object Tracking with markers" (3 strongly agreed and 5 somewhat agreed), with a single "neither agree nor disagree response", and 3 participants said they somewhat disagreed. Alternatively, only 3 participants only somewhat agreed or strongly agreed with the statement "I could align the object with satisfactory quality using Manual Object Placement." There were 4 neither agree nor disagree responses, and 5 somewhat disagree responses for the same question. The mean response scores (where

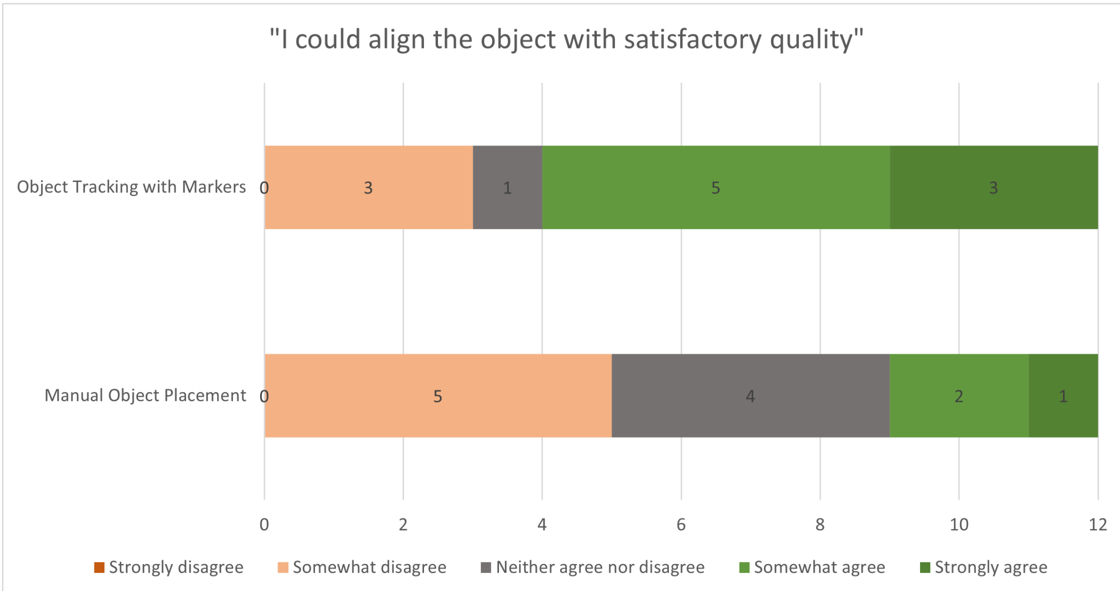

Figure 5.2: The responses to the statement ”I could align the object with satisfactory quality using the given method”

the Likert Scale is translated to a 1-5 scale) were 3.67 for object tracking and 2.91 for manual object placement. These results were found to be statistically significant ($p < .01$). The results of this question are displayed in Figure 5.2. The other Likert-Scale Responses with regard to alignment can be seen in the appendix.

The two other statements participants were asked to provide their agreement with were ”It was easy to achieve satisfactory alignment with the given method” and ”It was efficient to achieve satisfactory alignment with the given method”. For the first statement, both object tracking and manual object placement received the 6 strongly agree or somewhat agree responses (both receiving 1 strongly agree and 5 somewhat agrees). However, manual object tracking received more negative responses, getting 1 strongly disagree and 4 somewhat disagree responses. Object tracking received 2 somewhat disagrees, and the rest of the responses were ”neither agree nor disagree.” The responses to the question of efficiency were slightly more varied. Eight participants indicated they either strongly agreed or somewhat agreed (3 and 5 respectively) that alignment was efficient with object tracking, compared to 6 somewhat agree responses for manual object placement. There was a single ”neither agree nor disagree” response for both, and object tracking received 3 somewhat disagree and strongly disagree responses (2 and 1 respectively), compared to manual object placement’s 5 (3 and 2 respectively). The mean scores were 3.42 and 3.58 respectively for object tracking, and 3.08 and 2.92 for manual object placement. These results were also statistically significant ($p < .05$ and $p < .005$)

5.2.2 Scaling

Generally, the users also preferred physical markers for object scaling. Ten indicated it was their preference in the forced-choice question with 2 indicating they preferred the manual scaling methods. One of the nine however indicated in the follow-up question that they actually had no preference between the two methods, but felt that with some training they would likely prefer the scaling methods which came with physical markers (a menu with sliders). One user indicated that the separate menu was a benefit as they could slide to their desired value and check that the object looked correct after the fact. With the manual method, they felt they needed to both look at the menu and the object as they scaled it to get the desired results and this produced frustration as they could not focus on two things at once. Another felt it was significantly easier to achieve desired values when using the sliders instead of the pinch and stretch or independent scaling features. Another participant appreciated the scale sliders explicitly separating the concerns of moving and scaling the objects, as when pinching and stretching one can still move their hands and thus the object as well.

Only one participant elected to elaborate on why they preferred the manual scaling approach. The participant indicated that they felt the sliders were ”finicky” and took a longer time to use than the pinch and stretch or independent scaling options. The other participant mentioned scaling in their response to the question about alignment, however, saying ”It was easy to calibrate the dimensions”. Additionally, one of the participants who said they preferred the slider method mentioned that they would have liked using the sliders for manual manipulation. They said a method that separates the dimensions would have been useful and likely diminished their preference toward the other method.

Likert-Scale Repsonses

In the scaling section of the exit survey, 10 participants indicated that they either strongly agreed or somewhat agreed (5 and 5 respectively) with the statement ”I could scale the object to satisfactory quality using the Object Tracking with Markers”, compared to 5 (1 and 4 responses) for manual object placement. The negative responses skewed more toward manual object placement, with it receiving 5 strongly disagree or somewhat disagree responses (2 and 3), and object tracking receiving 2 somewhat disagree responses. Object tracking received no neither agree nor disagree responses and manual placement received 2. The mean scores for these responses were 4.08 for object tracking and 2.91 for manual object placement. These results are statistically significant ($p < .0001$). The results of this Likert-Scale Response can be seen in Figure 5.3. Graphs of the responses to the other scaling statements can be seen in the Appendix.

For the two statements following, ”It was easy to achieve a satisfactory scaling using the given method” and ”It was efficient to achieve a satisfactory scaling using the given

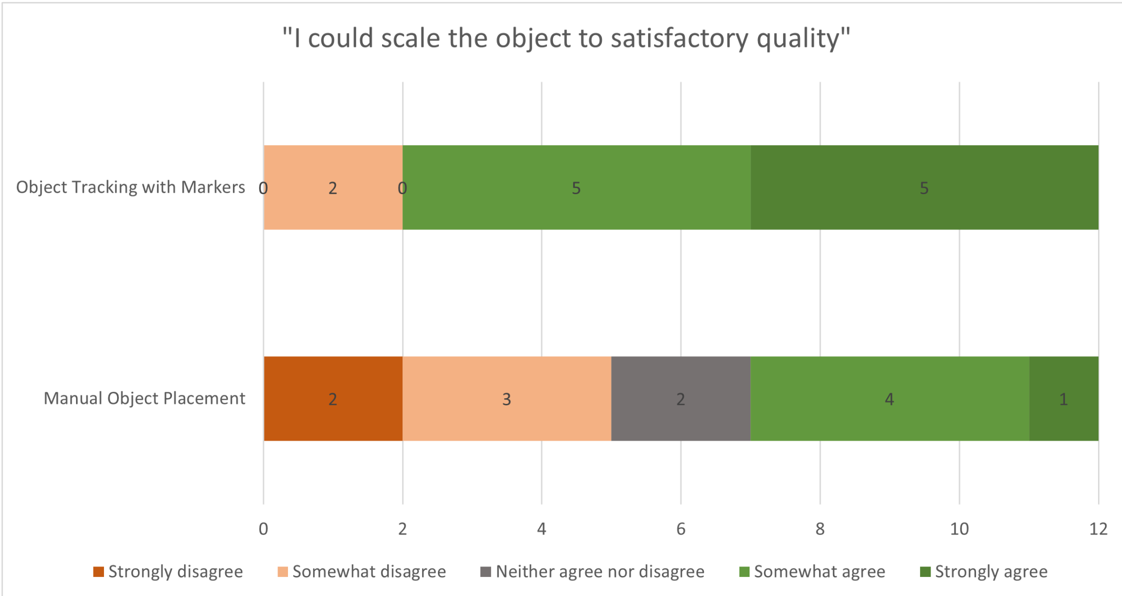

Figure 5.3: The responses to the statement "I could scale the object with satisfactory quality using the given method"

method", object tracking once again had several more strongly agree and somewhat agree responses than manual object placement. For the first statement, it received 6 strongly agree and 4 somewhat agree compared to manual object placement's 1 and 2. For the second, it received 9 total (4 and 5 respectively) while manual object placement received 5 total (2 and 3 respectively). For the other possible responses, object tracking received 1 of each, while manual object placement received 4 strongly disagree responses, 2 somewhat disagree responses, and a single neither agree nor disagree. The mean scores for these responses were 4.08 and 3.83 respectively for object tracking, and 2.58 and 2.75 respectively for manual object placement. The differences were also found to be statistically significant ($p < .0005$ and $p < .005$).

Chapter 6

Discussion

6.1 Quantitative Implications

The quantitative data indicates faster completion times when using object tracking with markers for 3 of the 4 tasks presented to participants. The speed differences come down primarily to users having an easier time with the scaling of objects, but alignment time also contributed to this. Users often struggled with manual object manipulation in tasks where objects did not need to be scaled equally in three dimensions. The sliders were often easier for the user to manipulate, and even when they had a difficult time with them, their struggles primarily centered around getting a given slider to the precise value they desired, which was a simpler problem to solve than attempting to scale to different sizes in multiple dimensions at once. Users who had challenges with independent scaling often struggled to maintain a scaling in one dimension while adjusting another. Additionally, those who had better results with it often still scaled one dimension at a time and did so with slow and intentional motions of their hands. Even those who preferred to use independent scaling still relied on separating degrees of freedom when scaling in order to perform the task as efficiently as possible. In regard to aligning objects, it was just as fast to move a digital object to a physical one as it was to move two physical objects for most, but once the object was placed, the manual object placement tasks often required the user to make fine adjustments to the manipulated object. When performing the object tracking tasks, these fine adjustments often did not need to be made, or if they did need to be made, it took less time to make a small adjustment to the tracked marker.

However, it is important to acknowledge that task completion times for all tasks were relatively high. Many conditions took the users over one minute to complete, and this was the case for all conditions on average. This is in conjunction with the fact that the study consisted almost entirely of AR/VR novices and beginners. This could have affected task completion time and the results found here may not generalize to AR/VR experts. That said, the one participant with substantial AR/VR experience did not have substantially different completion times from the means, and in fact, was often slower than the mean times. This likely implies a learning curve for both methods and that different results may be measured if users are given more time to familiarize themselves with the system.

6.2 Qualitative Implications

As previously stated, the results from all of the Likert Scale questions were statistically significant. Additionally, the mean scores for object tracking with markers were higher than that of manual object placement. This implies that in both the case of object alignment and in the case of object scaling the latter was an easier method for users to use. It is worth noting, however, that all but one participant indicated having either no experience (6) or only some experience (5) using AR/VR applications. This means the discussion of the results may not generalize across all AR/VR users and is an important consideration when discussing this data.

6.2.1 Object Alignment

Eight of the 12 participants indicated a preference for object tracking as their alignment method of choice, and this is corroborated by the results of the Likert Scale questions. Users indicated they could better achieve a satisfactory alignment by using the markers, and that it was easier and more efficient to do so with the markers. This all implies the use of physical markers to align digital objects with the real world is preferential to merely attempting to place the objects by hand.

There were two big themes that likely produced these results. The first of which is that perfecting rotation and ensuring flat lies with a given surface was difficult without the markers. One user stated "It's much more intuitive to place physical objects, since I've lived in the physical world my entire life. I haven't had even close to as much experience manipulating objects in AR, so it was much harder to align things." Users did not need to put in much effort to achieve quality alignments, as they merely needed to lay targets where they wished for them to be, and the 3D objects would follow. One user also had difficulty with alignment due to a shaking hand, this issue was mitigated with the tracker, as it would remain still when placed down and could be manipulated when placed on a flat surface without worry of it clipping into the surface or floating above the surface. The second is that users already understood how to manipulate physical objects. Re-examining the first quote, we can see that object tracking is an intuitive metaphor for object control in AR. One needs to be able to manipulate physical objects in order to get through everyday life, and participants were able to rely upon that intuition to efficiently and easily complete the alignment portion of their tasks.

An additional observation is that multiple (4) participants attempted to grasp the digital objects when using object tracking with markers. Though this can potentially be attributed to confusion due to having to switch between control metaphors for each task more investigation ought to be done than to blame it solely on confusion due to the different control methods provided. This likely implies it may be beneficial to find a way to allow for the manipulation of those rendered objects (merely making them manipulable is a trivial task) by grasping them in AR. One could imagine it may be worthwhile to make it so that one could handle rotation or scaling by grasping and manipulating the object, and then pressing a button or using a voice command to make it return to its original place attached to the image target which caused it to render.

6.2.2 Object Scaling

Ten of the 12 participants indicated a preference for object tracking as their scaling method of choice. This is compounded by the Likert Scale results indicating the participants could achieve more satisfactory scaling than the manual methods, found it easier to achieve that scaling, and found it more efficient to achieve that scaling. Accordingly, it is reasonable to say this scaling method is an improvement over the ”standard” implementation

Those findings correspond with both the notes taken during the study and the open responses provided in the post-exit survey. There were two primary reasons for the preference toward the object tracking implementation of scaling, the separation of degrees of freedom in scaling and its relative improvement in precision over the manual methods. Participants pointed out that scaling each dimension separately made it significantly easier to achieve the desired values without accidentally modifying the scale values of other dimensions. This makes sense, unlike translation and rotation tasks, scaling often does not need to be done in all 3 dimensions, and when it does, the required scales are often different in each dimension. Because of this even though it may take more time, separating the scaling along each dimension is a practical means of implementing the interaction and allows users to interact in a way that is useful and intuitive to them. Additionally, users are familiar with sliders from their use in other applications such as volume sliders. This translation of knowledge makes the UI clear to them and allowed for easy intuition of the scaling menu for most users. This method being more precise also greatly benefited it. Users were able to reliably and quickly move the sliders to the desired values (partially due to the rounding mode being enabled). Even when one participant said that when it may make more sense to use the pinch-and-stretch method for uniform scaling, they still liked the sliders as they made small-scale changes easier to perform.

The only consistent issue users had with scaling using object tracking with markers was that they sometimes had problems grasping the sliders on the menu. The primary issue was with depth perception, both of the menu itself and of the sliders. In addition to this one participant had trouble interacting with the scaling menu as they thought they were using a dial instead of a slider and attempted to twist the slider once they had pinched it. The participants who preferred manual scaling also had a difficult time getting the sliders to stay on the correct value when releasing the slider. This created a feeling of imprecision, which was also present for some of the participants who still indicated they preferred using the sliders as their scaling method.

Those who preferred manual scaling did not present much information on why they preferred it. One user indicated that using their hands to scale the objects felt more intuitive. They also said that they felt it was a more consistent experience, they had trouble using the sliders, and at no point with scaling with the manual object manipulation were they hindered by it. It is worth noting, however, that this participant struggled with both control methods, particularly in the case of alignment. In their Likert scale responses, they indicated they somewhat disagreed with all provided statements for both control methods. The other user neglected to elaborate on why they preferred the manual scaling in the question asking why they chose the method they preferred, but when answering the final question they said that scaling was able to be done smoothly when they moved slowly and gently. They did not feel they were able to achieve this when using the sliders, which is why they preferred the method they did. It is also worthy of note that during their trials this participant did become visibly frustrated with the independent scaling mode, despite indicating they preferred manual object scaling.

Manual scaling presented multiple obstacles for users, however. Firstly, the pinch-and-stretch metaphor is inherently limited as it only allows for equal scaling across all three dimensions. Though users had little trouble understanding and using the control method, not being able to use it for tasks that demanded non-uniform scaling was both a frustration and a point of issue for them, as it was the simpler and preferred method. Many participants did not like independent scaling, with one saying "Independent scaling sucks!", and another reporting "Independent scaling did not feel consistent in how sensitive it was to hand movements". This is due to the fact that the value of the scale change is relative to the displacement of the hand when performing the operation, if the hand is moved a few inches, the object is scaled to match this movement. This means large motions are needed to scale the objects to large sizes and precise motion becomes difficult when the objects are small. The participants who disliked independent scaling generally mentioned they preferred being able to separate the degrees of freedom when scaling objects, and felt it was much simpler too. This indicates that although 3 DOF translation and 3 DOF rotation are desirable, 3 DOF scaling is significantly less so. In addition to the frustrations with independent scaling, neither metaphor was well suited to scaling objects down, particularly when the objects were already small. The second condition for task 1 was to shrink the provided cube down to $\frac{1}{3}$ of its original size, and regardless of which control metaphor was used participants had a difficult time as they struggled to grasp a digital object that small, and in the case of pinch-and-stretch, to push their hands close enough together to meaningfully scale the object down.

6.3 Limitations

6.3.1 Print Quality of Markers

The print quality of the markers affects the ability of Vuforia to track them. There are multiple factors that can affect the Vuforia Engine's ability to track a marker. These include the type of printer and paper used to create the object, any creasing or bending of the marker, and to a lesser extent, the physical size of the markers. Though no participants were provided with creased Image Targets, they were printed onto standard 8.5x11in US letter paper, which would bend when picked up from the in front of participants. This can cause the engine to temporarily lose track of the marker, and negatively affect the user experience. Similarly, the printer used to create these markers produced relatively glossy images. Though the markers still work in most lighting conditions, bright fluorescent light reflects off of the markers causing glare and further tracking issues. In tasks 4 and 2 lighting conditions sometimes made it difficult to keep the targets in focus to render the desired content.

The size of the markers is also another limitation, the image target needs to be in focus in order for the object to initially be detected. This meant that the user often needed to hold the marker near their head in order to get the Hololens to render the pop-socket for the task, and then once the object was rendered it would be able to be tracked and manipulated appropriately. Another example of size limitations came in task 4. Although the targets used could have been made smaller, there is still a limitation to the size they can viably be before the Hololens camera will not be able to keep them both in focus (and thus render the relevant content coming with them). However, the size of the targets themselves imposes a limit on how close the two targets can be pushed together if either target becomes occluded by the other one can no longer use the occluded target (it is noteworthy that task 4 did not require moving the targets in order to bridge the gap, though one user did attempt to do this regardless). Particularly in the case of interactions requiring multiple targets, the developer needs to ensure the planned interaction uses both targets of the appropriate (but minimal) size, and targets that will be able to remain in focus from a variety of angles.

Print quality also comes into play when considering the types of interactions this study prioritized. In all cases, the users were asked to lay the target flat on the surface in front of them in order to complete their task. However, this is not the only way to interface with a marker. Though in the case of object alignment and verification, this is a reasonable method, it is by no means the only way to inspect an object and verify that it is where it is meant to be. In use cases where one is tasked with holding the marker in the air or wishes to examine the digital object from a variety of angles by tilting the marker, the quality of the printed image becomes much more important. If the image is glossy, this can cause Vuforia to render the object in the wrong place, or not render it at all. As such the interaction needs to be implemented in a manner such that it can handle losing the tracked marker for a period of time, have imprecise tracking be a non-issue, or ensure the user has a matte target such that any bright lighting becomes more diffuse and the object is easier to track. Particularly as the perspective on the tracked object begins to skew and the user tilts the target toward or away from the camera, features within the target become obscured. As fewer features are tracked it becomes more important to ensure those that are still visible are able to be tracked and tracked in a quality manner.

6.3.2 Confounding Variables

This work treats task completion time as a dependent variable contingent upon the conditions of Object Tracking with Markers and Manual Object Placement being enabled. However, task completion time is a measure of one's ability to both align and scale given objects, and the scaling implementations are different between the two control methods, as are the alignment methods. The differences in task completion time were a result of both of these factors, and a fairer comparison would be to independently verify the differences in time to align the object, and then again for the time to scale the object once aligned (or vice versa). Instead, it is unclear the degree to which either of those variables affects overall task completion time, all that can be verified is that there is a gap between the two control methods. A more thorough study would be needed to determine the effects of alignment and scaling separately and we leave this to future work.

6.3.3 Scaling with Tracked Objects

The object-tracking implementation of object scaling still relies on the AR camera's ability to track fine-grained input without a physical prop aside from the hand. As such although an image tracking system can resolve problems with object alignment and ensure digital objects are placed where the user wishes for them to be as well as placed there at the correct rotation, the same cannot be said for object scale. Coarse-grained scaling can be handled simply via UI buttons and/or voice commands (a method unexplored in the study but was implemented in an earlier version of this work). However, the solution of using sliding scales to handle object scaling is sub-optimal, as it takes an imprecise process in AR and turns it into a new, also imprecise process. Improvements could be made such as allowing the user to manipulate the C-D ratio of the slider, or clarifying what the feedback provided by the slider means. The current implementation of the slider plays a clicking sound as one moves the slider along its axis, and although this helps the user to understand the distance it has moved, there is no clear connection between what any individual click indicates. It is best used as an indication that one has successfully begun to manipulate the slider and that one can let go of the slider when it is in the desired place. Additionally, the menu containing the sliders could independently be grasped, and users sometimes had issues grasping the sliders or grasping the menu instead of the sliders.

6.3.4 Manual Object Scaling

Many of the complaints from users regarding scaling when in manual object manipulation mode centered around not being able to scale the object one dimension at a time. Though the independent scaling option allowed one to separately scale the objects in each dimension, it still scales in all three dimensions at the same time based on the motion of one's hand in each dimension. It is challenging to keep one's hand still in one, let alone two dimensions, a fairer comparison may have been an implementation that allows stretching in a single dimension while freezing the others.

6.3.5 Perspective

The perspective with which a user examines an object greatly affects how the user perceives the quality of their alignment. At multiple times during the study, the users elected to mark their task as complete after aligning a digital object using manual alignment despite the fact that the object was not actually aligned with the physical object they were trying to place it on. Depending on how the user views the object it can appear to be aligned when in fact it is not. This was particularly apparent in task 3, where the users were asked to place a triangular prism on a box in order to give it a roof, the users often did not get a truly flat lie, but it would appear as such from head-on. Moving around the object would make this apparent, but some users did not check or did not adequately adjust the object after the fact. This is likely partially due to the fact that the tasks in the study were predicated upon the user indicating completion when *the participants* were satisfied with the quality of the digital object's alignment with the physical object. This implies a degree of inherent imprecision in the ability to align objects due to hand tracking and lack of tactile feedback, and to an extent limits the findings with regard to the effectiveness of object alignment using manual object placement.

Chapter 7

Conclusions

As it currently stands, precise object manipulation and alignment in AR is difficult, particularly without placing limitations on the degrees of freedom of one's movements. To resolve this, we proposed and designed a control scheme using tracked objects and markers to handle the process of precisely aligning and scaling AR content within the physical environment. It was then tested against a control scheme representing a ”standard” implementation of these same controls and found that it was more efficient in 3 of the 4 tasks performed by study participants and that they agreed that object tracking with markers was more effective, easier, and more efficient than manual object placement.

There is substantial room for further study. The degree to which the user can interact with objects rendered via object tracking is intentionally limited. There is still significant room to implement new interactions leveraging these markers, as well as hybrid interactions allowing for more complex uses of the markers than just using them for positioning and rotating digital objects. Even something as simple as using the targets to initially align, rotate and scale objects followed by other operations is unsupported, and this would make for a stronger and more useful visualization application. This would also allow for the use of larger targets which are easier to handle and keep in focus, before removing them to actually verify one's alignment of the objects.

Taken as a whole, this work represents an initial showing that tracking real-life objects is a meaningful and intuitive way for users to better align digital objects with other real ones. Participants in our study both indicated a preference toward this control method, and their task completion times also indicate increased efficiency when using the markers over manually attempting to place and scale digital objects. This, in conjunction with the minimal supplies needed to create additional markers, make it ideal for tasks such as visualizing models and with some additional development, placing interactable AR content in one's environment.

Bibliography

[1] Hiroshi Ishii and Brygg Ullmer. "Tangible bits". In: *Proceedings of the ACM SIGCHI Conference on Human factors in computing systems*. ACM, Mar. 1997. DOI: 10.1145/258549.258715.

[2] Doug Bowman et al. "3D User Interfaces: New Directions and Perspectives". In: *IEEE Computer Graphics and Applications* 28.6 (2004), pp. 20–36. DOI: 10.1109/mcg.2008.109.

[3] Hannes Kaufmann and Dieter Schmalstieg. "Mathematics and geometry education with collaborative augmented reality". In: *Computers amp; Graphics* 27.3 (2004), pp. 339–345. DOI: 10.1016/s0097-8493(03)00028-1.

[4] *Nintendo Wii*. 2006.

[5] Youngmin Park and Woontack Woo. "The ARTable: An AR-Based Tangible User Interface System". In: *Technologies for E-Learning and Digital Entertainment*. Springer Berlin Heidelberg, 2006, pp. 1198–1207. DOI: 10.1007/11736639_150.

[6] Feng Zhou, Henry Been-Lirn Duh, and Mark Billinghurst. "Trends in augmented reality tracking, interaction and display: A review of ten years of ISMAR". In: *2008 7th IEEE/ACM International Symposium on Mixed and Augmented Reality*. IEEE, Sept. 2008. DOI: 10.1109/ismar.2008.4637362.

[7] Trien Do and Jong-Weon Lee. "A Quick Algorithm for Snapping 3D Objects in Augmented Reality". In: *2009 International Symposium on Ubiquitous Virtual Reality*. Vol. 39. 3. IEEE, 2010, pp. 377–386. DOI: 10.1109/isuvr.2009.23.

[8] Trien V. Do and Jong-Weon Lee. "3Darmodeler: A 3D Modeling System In Augmented Reality Environment". en. In: (2010). DOI: 10.5281/ZENODO.1082871.

[9] Hyeongmook Lee, Mark Billinghurst, and Woontack Woo. "Two-handed tangible interaction techniques for composing augmented blocks". In: *Virtual Reality* 15.2-3 (June 2010), pp. 133–146. DOI: 10.1007/s10055-010-0163-9.

[10] Kangdon Lee. "Augmented Reality in Education and Training". In: *TechTrends* 56.2 (Feb. 2012), pp. 13–21. DOI: 10.1007/s11528-012-0559-3.

[11] Christian Weichel et al. "MixFab". In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, Apr. 2014. DOI: 10.1145/2556288.2557090.

[12] Xiang 'Anthony' Chen et al. "Reprise". In: *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*. ACM, Oct. 2016. DOI: 10.1145/2984511.2984512.

[13] Benjamin Nuernberger et al. "SnapToReality". In: *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. ACM, May 2016. DOI: 10.1145/2858036.2858250.

[14] *Oculus Rift*. 2016.

[15] *Pokemon Go*. 2016.

[16] *Unity - Game Engine*. Nov. 8, 2017.

[17] *Blender - a 3D modelling and rendering package*. Amsterdam, 2018.

[18] F A Purnomo et al. "Implementation of Augmented Reality Technology in Sangiran Museum with Vuforia". In: *IOP Conference Series: Materials Science and Engineering* 333 (Mar. 2018), p. 012103. DOI: 10.1088/1757-899x/333/1/012103.

[19] Devamardeep Hayatpur et al. “Plane, Ray, and Point”. In: *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology*. ACM, Oct. 2019. DOI: 10.1145/3332165.3347916.

[20] Xuyue Yin et al. “An Automatic Marker–Object Offset Calibration Method for Precise 3D Augmented Reality Registration in Industrial Applications”. In: *Applied Sciences* 9.20 (Oct. 2019), p. 4464. DOI: 10.3390/app9204464.

[21] Mayra D. Barrera-Machuca, Alvaro Cassinelli, and Christian Sandor. “Context-Based 3D Grids for Augmented Reality User Interfaces”. In: *Adjunct Publication of the 33rd Annual ACM Symposium on User Interface Software and Technology*. ACM, Oct. 2020. DOI: 10.1145/3379350.3416190.

[22] David Englmeier et al. “A Tangible Spherical Proxy for Object Manipulation in Augmented Reality”. In: *2020 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*. IEEE, Mar. 2020. DOI: 10.1109/vr46266.2020.00041.

[23] Danilo Gasques, Alexander Scheurer, and Pedro Lopes. *danilogr/AR-Marker-Generator v1.0*. 2021. DOI: 10.5281/ZENODO.4466705.

[24] Mojtaba Noghabaei and Kevin Han. “Object manipulation in immersive virtual environments: Hand Motion tracking technology and snap-to-fit function”. In: *Automation in Construction* 124 (Apr. 2021), p. 103594. DOI: 10.1016/j.autcon.2021.103594.

[25] *Microsoft Mixed Reality Tookit*. 2022.

[26] *Vuforia Engine 9*. 2022.

[27] *PS VR*. 2023.

Appendices

Appendix A

Appendix

A.1 Data

Linked below is a view-only Google sheet used to calculate p-values and means for all data. There is also a link to a downloadable Excel spreadsheet used to make the graphs.

https://bit.ly/3Kw74R8

https://bit.ly/3KFHRUn

A.2 Additional Screenshots

Below are the initial conditions for the manual versions of each task.

A.3 Additional Graphs

Here are graphs of the responses to the 4 other Likert-Scale statements.

Figure A.1: Task 1

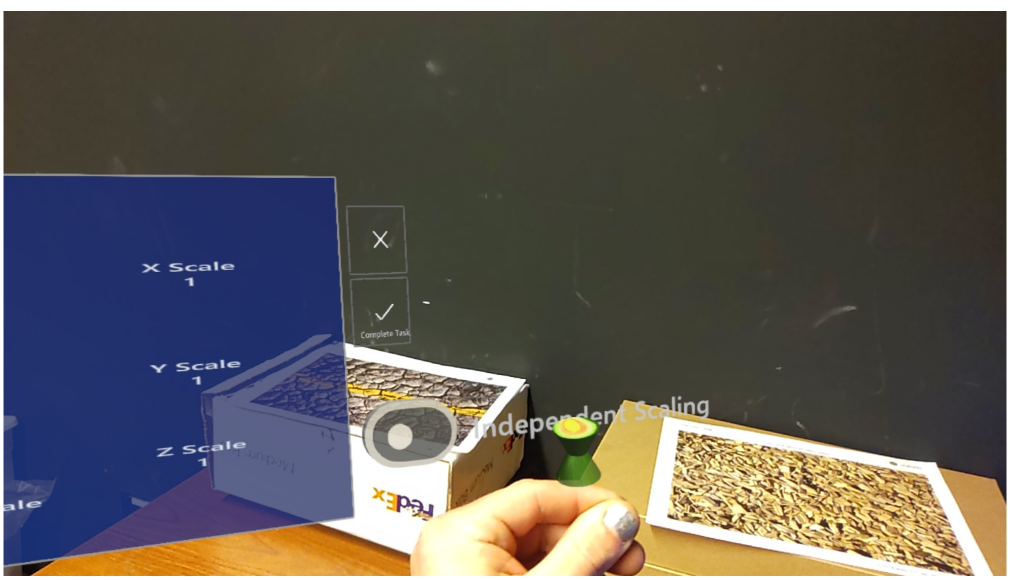

Figure A.2: Task 2

Figure A.3: Task 3

Figure A.4: Task 4

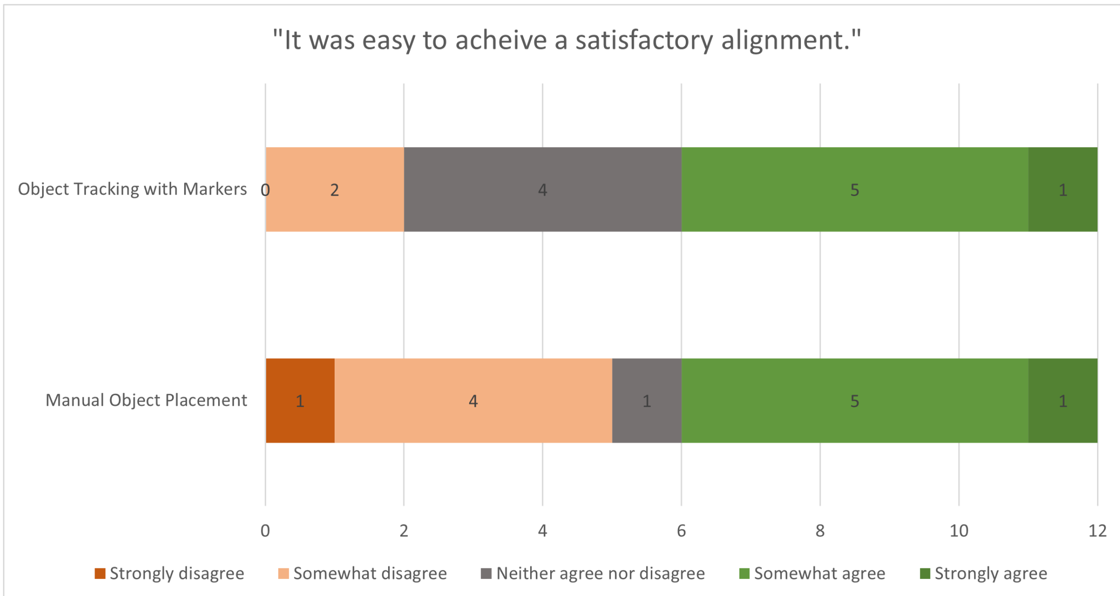

Figure A.5: Responses for ease of alignment

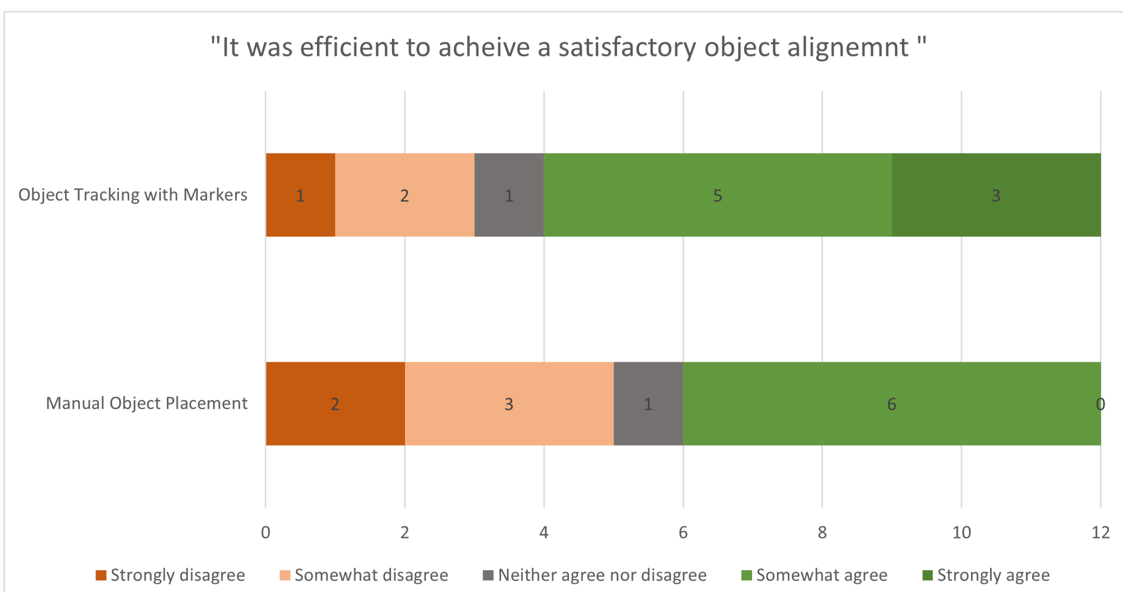

Figure A.6: Responses for efficiency of alignment

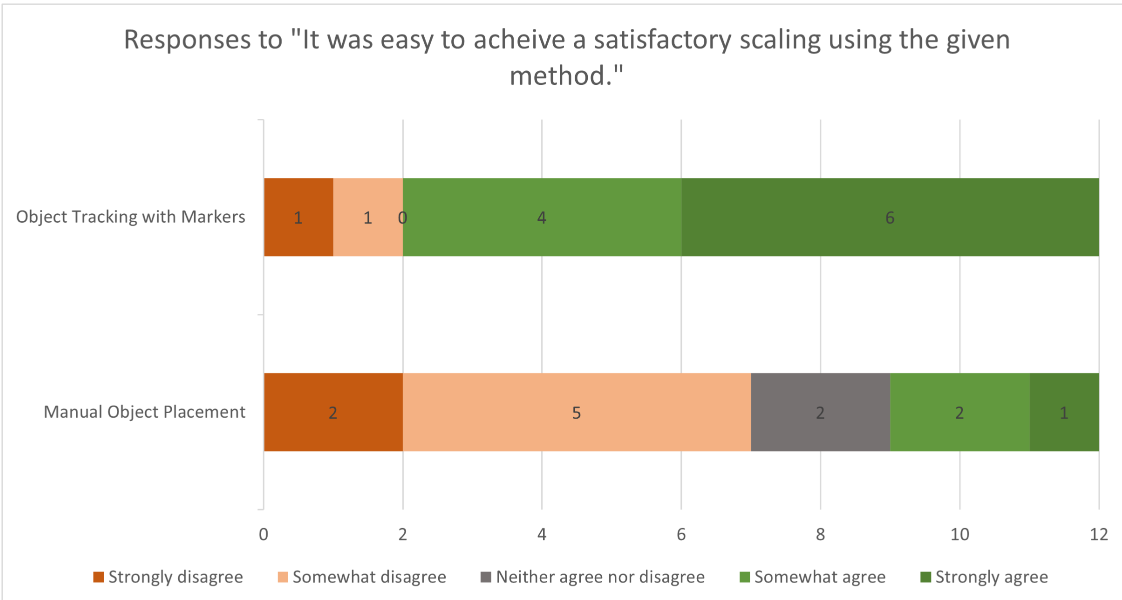

Figure A.7: Responses for ease of scaling

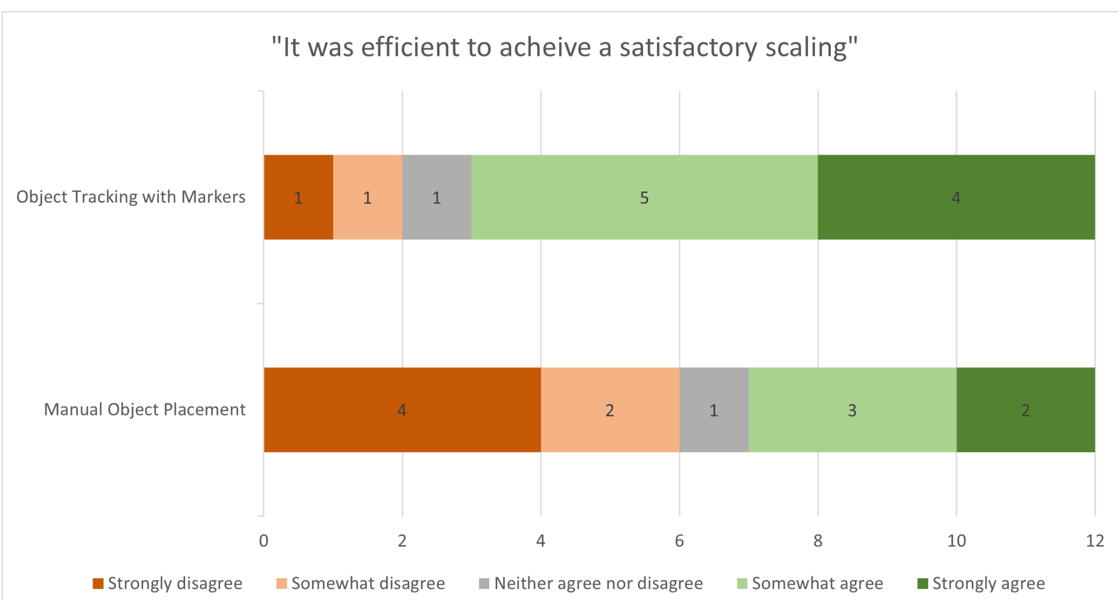

Figure A.8: Responses for efficiency of scaling