An Evaluation of Bimanual Gestures on the Microsoft HoloLens

Nikolas Chaconas,[*] Tobias Höllerer[†]

Computer Science Department
University of California, Santa Barbara

ABSTRACT

We developed and evaluated two-handed gestures on the Microsoft HoloLens to manipulate augmented reality annotations through rotation and scale operations. We explore the design space of bimanual interactions on head-worn AR platforms, with the intention of dedicating two-handed gestures to rotation and scaling manipulations while reserving one-handed interactions to drawing annotations.

In total, we implemented five techniques for rotation and scale manipulation gestures on the Microsoft HoloLens: three two-handed techniques, one technique for one-handed rotation and two-handed scale, and one baseline one-handed technique that represents standard HoloLens UI recommendations. Two of the bimanual interaction techniques involve axis separation for rotation whereas the third technique is fully 6DOF and modeled after the successful ”spindle” approach from 3DUI literature. To evaluate our techniques, we conducted a study with 48 users. We recorded multiple performance metrics for each user on each technique, as well as user preferences. Results indicate that in spite of problems due to field-of-view limitations, certain two-handed techniques perform comparatively to the one-handed baseline technique in terms of accuracy and time. Furthermore, the best-performing two-handed technique outdid all other techniques in terms of overall user preference, demonstrating that bimanual gesture interactions can serve a valuable role in the UI toolbox on head-worn AR devices such as the HoloLens.

Keywords: Bimanual, two-handed, gestures, object manipulation, rotation, scale, evaluation, user study, augmented reality, HoloLens

Index Terms: H.5.1 [Information Interfaces and Presentation]: Multimedia Information Systems—Artificial, augmented, and virtual realities; H.5.2 [Information Interfaces and Presentation]: User Interfaces—Input devices and strategies;

1 INTRODUCTION

Augmented Reality is a convenient UI paradigm for creating annotations of real word objects. The Microsoft HoloLens is a device well suited to this task as it offers head and hand tracking, as well as spatial mapping of physical operation environments. Previous work exists for assessing the most accurate and preferred methods for creating one-handed annotations on the HoloLens [1]. However, users commonly lack the ability to change the orientation or size of an annotation without re-drawing the annotation. Furthermore, in an environment with only one-handed interactions, the addition of one-handed scale and rotation gestures would require some method for switching between annotation-drawing and annotation-manipulation modes. Alternatively, as our work explores, two hands can be used for spatial manipulation tasks, and one hand can be reserved for drawing annotations. This way, all visual indicators needed for rotation and scale could be hidden when only one hand is in the air, allowing annotations to not be obstructed. The Microsoft HoloLens currently only recommends one-handed gestures, and discourages the development of two-handed gestures on their developer forums [4], with the biggest concern about two-handed gestures being the limited hand tracking area in front of the device. Despite this, countless literature has demonstrated that bimanual interactions can outperform uni-manual interactions in 3D manipulation tasks [2, 6, 12, 19], providing strong motivation for the exploration of bimanual gestures on the HoloLens. This work explores the feasibility and justification of developing two-handed gestures on the HoloLens, contributing four different approaches for manipulating drawn annotations using two-handed gestures and comparing them to a standard one-handed manipulation method on the HoloLens.

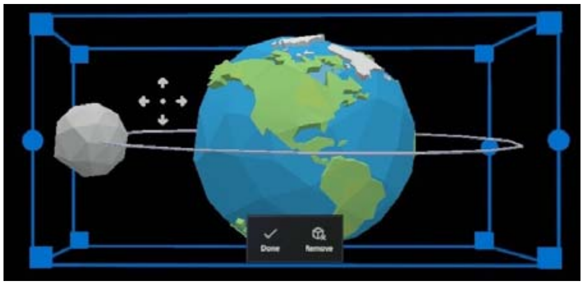

Figure 1: Object manipulation within Hologram app currently provided on the HoloLens. Dragging the corner boxes scales the Earth-Moon configuration and dragging the round wireframe nodes rotates the object around the vertical axis.

To evaluate the design space of two-handed interactions on the HoloLens, we conducted a within-subjects user study with 38 participants comparing the time and accuracy of performing each gesture to complete simple reference tasks. As a baseline comparison, we also implemented a technique similar to the one-handed Wireframe Cube technique currently in use on standard HoloLens applications (cf. Fig. 1). The Wireframe Cube technique as implemented, e.g., in the default Hologram viewer, only allows for rotation along one axis (yaw), thus in order to allow for a fair comparison between this technique and our proposed two-handed techniques, we modified the Wireframe Cube and added the possibility for rotation about any axis. In our results, we found that overall the Wireframe Cube technique afforded more accurate manipulation than the other techniques by a small margin, and that there wasn't a significant difference in timing among the best performing techniques. One two-handed technique, our novel ”Hands Locking into Gesture” technique, was most preferred by users compared to all other techniques, including Wireframe Cube, and showed no significant difference in terms of performance compared to the Wireframe Cube technique. Our results indicate that the possibility for a two-handed interface on the HoloLens is not only feasible, but can indeed be a valuable UI option according to user feedback and performance.

2 RELATED WORK

A main motivation for this work is the exploration of two-handed object manipulation options on the Microsoft HoloLens for the purpose of more convenient annotation placement. We discuss related work in the area of such annotation placement, general bimanual interactions in AR and VR, and specific implementation efforts on the HoloLens.

---

[*] e-mail: nikolas.chaconas@gmail.com

[†] e-mail: holl@cs.ucsb.edu

2018 IEEE Conference on Virtual Reality and 3D User Interfaces
18-22 March, Reutlingen, Germany
978-1-5386-3365-6/18/$31.00 ©2018 IEEE

2.1 3D Annotations

Existing work evaluated the use of one-handed annotation drawing on the HoloLens [1]. Two-handed manipulation gestures would allow for a convenient mode-less annotation authoring environment where one-handed drawing gestures would be dedicated to the creation of annotations, and two-handed scale and rotation gestures would be dedicated to the manipulation of created annotations.

2.2 Bimanual Interactions in AR/VR

There has been extensive previous effort in creating environments for object manipulation in both Augmented and Virtual Reality.

2.2.1 Exploring DOF

In choosing the best two-handed interactions, we wanted to explore the differences between both free-form six-degree-of-freedom manipulation, as well as axis-by-axis degree-of-freedom (DOF) separation for spatial manipulation tasks. Some research recommends higher degrees of freedom in performing object manipulation tasks [2, 3, 15]. Schultheis et al. studied three different modes of object manipulation on a 2 DOF (mouse), 6 DOF (wand), and 6 + 6 DOF two-handed interface (THI). Their results indicated that although their 6 + 6 THI had slightly longer training times than the other two interfaces, it significantly outperformed both the mouse and wand in terms of task completion time. Furthermore, the wand also greatly outperformed the mouse, leading them to conclude that many-DOF interfaces have an intrinsic advantage over a 2-DOF counterpart in fundamental 3D tasks. [15].

Mendes and colleagues, however, found that DOF separation can actually lead to improved results for accurately placing an object in a virtual environment [11]. They also reported that although full DOF separation led to higher precision in object manipulation tasks, it also led to longer completion times. More description on each of our implemented two-handed techniques and their DOF is explained in detail in Section 3.

2.2.2 Centroid Anchored Spindle Implementations

Mapes and Moshell contributed a two-handed virtual object manipulation interface, including the original ”Spindle” technique for 6DOF object rotation and scaling [9]. Without citing specific questionnaire results, Mapes and Moshell’s work reports that users preferred two-handed rotational gestures with 5DOF + Scale over one-handed rotational techniques [9]. Several modifications of the Spindle technique exist [6] to make it a full 6DOF approach. Song et al. explored a handle bar metaphor which they enacted using the Microsoft Kinect. A virtual handle bar was placed through the target object centroid and fixed to the object when the user’s hands were in closed fists. Like the Spindle technique, the object could be rotated about the y and z axes by moving the fists and thus manipulating the handle bar. This method only allowed for rotation about the x axis by a technique which they called ”peddling” [17]. Peddling allowed for an incremental pitch rotation by a movement of both hands about the y and z axes simultaneously in one direction. They speculated that although this peddling motion enabled pitch rotation, it may not be immediately intuitive for uninitiated users [17]. They also presented what they called a ”constrained rotation”. With this technique, one hand could be stationary, and the other hand could circle as if winding a crank about the x axis in order to perform pitch rotation. Although more intuitive than the peddling motion, this provision requires a mode switch, long recognized as a potential source of errors and confusion [13].

We base our implementation for 6DOF manipulation (”Spindle and Raise”) on the Spindle + Wheel technique [2]. Mendes and colleagues posit that DOF separation improves accuracy for placing objects [11], and Cho’s work claims that the performance of a DOF gesture is dependent on the actual DOF needed by the manipulation task [2].

Other research on user hand interaction has been done using gloves [5, 8, 18], exploring the use of different two-handed gestures in immersive environments. Although gloves allow for more efficient and reliable hand gesture detection than other HCI techniques, a user could find gloves uncomfortable or restricting to their hand movement.

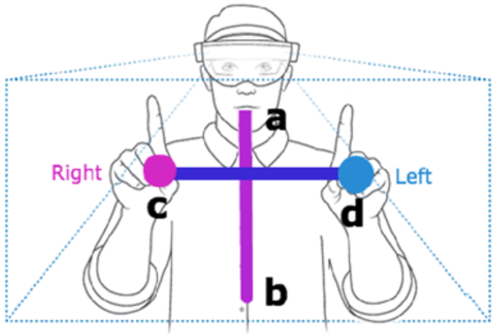

Figure 2: Distinguishing right and left hand based on cross products between gaze direction $\overrightarrow{b-a}$ and hand positions $\overrightarrow{c-a}$ and $\overrightarrow{d-a}$.

2.3 Bimanual Interactions on the Microsoft HoloLens

Bimanual spatial interactions have not yet been well formally evaluated on the Microsoft HoloLens. Bill McCrary explored two-handed manipulations on the HoloLens and developed a method of two-handed scale and rotation of objects with the following properties [10]:

- Two hands visible, one pinched: rotate the object
- Two hands visible, both pinched: scale the object

He found, however, that after a period of extended use it became uncomfortable to have both hands up and in the correct positions all the time. Due to this, his final iteration did not involve bimanual techniques, but instead a voice-activated selection of different modes. We expected similar ergonomic limitations, but argue that there will be situations where bimanual interactions will be natural and effective (such as for quickly adjusting scale and orientations of annotations). Understanding the potential and limitations of such gestures is important.

Our work thus explores the addition and comparison of multiple bimanual manipulation gestures to the Microsoft HoloLens, without the use of external trackers, and compares them to each other and reference one-handed interactions.

3 SYSTEM AND TECHNIQUES

Initial attempts to incorporate hand segmentation in OpenCV similar to [7] proved to run too slowly for real-time application on the HoloLens. Thus, hands were tracked using the Microsoft HoloLens API for hand tracking. Using this API, events can be registered for each hand, and right and left hands can be distinguished by comparing the cross products between the gaze direction and the position vectors of the tracked hands (see Fig. 2). We ignored the y-axis components in the cross product computations. Negative cross product results were classified as right hands, and positive ones were classified as the left hand. Both hands being to the right or left of the user’s gaze are dealt with by looking at the magnitude of the cross products to distinguish between right and left hand. If hands crossed, they would be reassigned as left and right based on this computation, causing the right hand to be assigned as the left, and vice versa. We felt that keeping the initial correct assignment of right and left hand through a hand cross, while potentially feasible, might be confusing, as a right indicator fixed on the target object would now correspond to the hand positioned leftmost of the body, and similarly with the left hand. Generally, when hands are crossed, the HoloLens loses tracking of hands during the crossover.

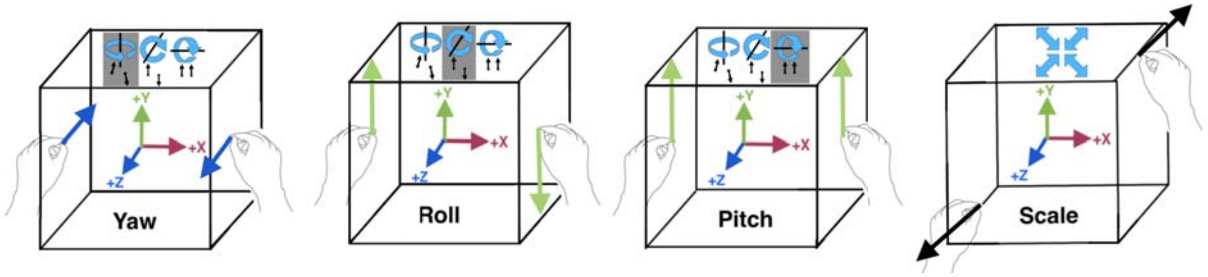

Figure 3: Hands Locking Into Gesture: example yaw, roll, pitch, and scale gestures. Hands could be moved in opposite directions, too.

All gestures committed by the user were scaled polynomially (4^{th} degree) when applied as a rotation or scaling manipulation to the object. As a result, a 180° rotation can be achieved with a relatively small amount of hand movement. This was to reduce the amount of hand movements needed by the user, and to reduce the user crossing their hands.

The last step towards creating usable gestures was monitoring events for whether the hands found were in the ready state vs. the pinch state, allowing for the creation of four different two-handed techniques for rotation and scale gestures. For the most part, every manipulation involved the following three gesture stages (GS), which will be referred to frequently in the remainder of the paper:

1. Positioning the hands in the Microsoft ”ready position” (raised index finger, conveniently executed as ’L’ (left hand) and mirrored ’L’ (right hand) shapes) and then pinching the relevant hands (GS1)
2. Manipulate object through rotation or scale gesture (GS2)
3. Ending manipulation gesture due to an open hand or lost hands (GS3)

In the case of one-handed gestures, if the hand was lost, GS3 was invoked. In the case of two-handed gestures, if one hand was lost, the last known position of the lost hand would be used. This ensured that gestures were continued regardless of hand losses to achieve the highest usability. The only two-handed gesture that could not easily utilize this approach was the Spindle with Raise gesture, as the object transformation is directly based off of the line connecting both hands. Furthermore, with the Microsoft hand tracking API, once a hand was lost in the pinch state, it couldn’t be tracked again until entering the open state once again. However, any hands which enter the open state (interpreted as purposefully ending the two-handed interaction) will invoke GS3.

For all two-handed gestures, with both hands raised and tracked in the open position, indicators are visible on the object. The indicators on the right and left of the object are associated with the right and left hands, respectively. An indicator colored as green represents a tracked hand, red indicates that the hand has been lost. Yellow indicates that the hand is in the closed position. It should be noted that ultimately if these two-handed gesture techniques were integrated with one-handed ones, one hand being lost (when not within GS2) would switch to the annotation drawing mode.

The coordinate system we employed follows HoloLens standards, with the positive x axis pointing to the right of the user, the positive y axis pointing straight up, and the positive z axis pointing towards the user.

3.1 ’Hands Locking Into Gesture’ Technique

This technique involves full DOF separation, and the specific rotational gesture is chosen based on the direction in which the user moves their hands after pinching both hands. Once locked into a gesture, a user is not able to perform another gesture until GS3 is invoked.The user is shown indicators above the object demonstrating which ways the hands could move to invoke different rotations (Fig. 3).

3.1.1 Hands Locking Into Gesture: Rotation

To perform the rotation portion of this gesture, the user must start with both hands at the same position on the y axis. Upon placing hands in this position, the user will see an indicator with the available rotations. The following rotations were achievable:

- Yaw: Pinching both hands and moving them in opposite directions - one hand along the positive z axis, and the other hand along the negative z axis.
- Roll: Pinching both hands and moving them in opposite directions - one hand along the positive y axis, and the other hand along the negative y axis.
- Pitch: Pinching both hands and moving them in the same direction - both hands along the positive y axis, or both hands along the negative y axis.

3.1.2 Hands Locking Into Gesture: Scale

To perform the scale portion of this gesture, the user must start with their hands at different Y positions. They can then pinch both hands and move both hands away from each other and towards each other.

There are two distinguishing positions to invoke scale for Hands Locking Into Gesture: either the right hand can be at a larger Y position than the left hand, or vice versa.

We did consider a variant in which both scale and rotation gestures begin with the hands in the same position, rather than have a different starting position for each. However, it was difficult to accurately distinguish between the hands moving away from each other in a scale gesture vs. the hands moving apart performing a rotation gesture. Thus, separating starting positions turned out to be a more robust way to distinguish gestures.

Partially based on this insight, we designed the following overall technique, in which all axis rotations and scale are distinguished by different hand starting positions.

3.2 ’Hands Starting Positions’ Technique

Like Hands Locking Into Gesture, the Hands Starting Positions technique also involves full DOF separation. All rotational and scale gestures in this technique are performed depending on the starting hand position before GS1 is invoked. Similar to Hands Locking Into Gesture, this technique had indicators above the object which detailed which starting positions the hands should be in to invoke different manipulations (Fig. 4). As soon as the user’s tracked hands were determined to be in the correct positions for initiating the corresponding transformation, the indicator would be highlighted (even before pinching).

All positions can be inverted, i.e., there exist exactly two possible starting positions for yaw and scale, and four for pitch (cf. Fig. 4).

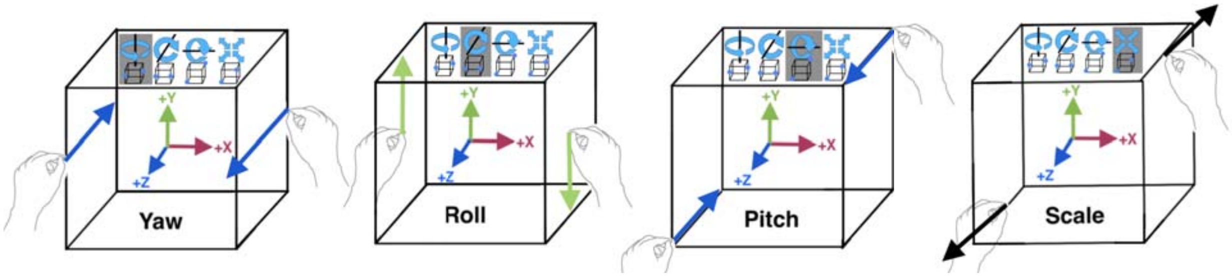

Figure 4: Hands Starting Positions: example pitch, yaw, roll, and scale gestures. Left and right hand position could also be inverted and Hands can be moved in opposite directions, too

There is only one starting position for roll (hands on the same Y and Z position, separated only in X).

To perform an axis rotation or scale operation, the user must start in the corresponding starting position for the respective operation:

- Yaw Rotation: Placing both hands in the "L" position at the same Y position, but different Z positions in front of the face, the user can then pinch both hands and move one hand along the positive Z axis, and the other hand along the negative Z axis to conduct the rotation.
- Roll Rotation: Placing both hands in the "L" position at the same Y position, and the same Z position in front of the face, the user can then pinch both hands and move one hand along the negative Y axis, and the other along the positive Y axis to conduct the rotation.
- Pitch Rotation: Placing both hands at different Y positions, and different Z positions in front of the face (i.e. on opposite corners of an imaginary cube, cf. Fig. 4), the user can then pinch both hands and move one hand along the positive Z axis, and the other hand along the negative Z axis.
- Scale: The scale portion of this technique is very similar to the scale portion of the Hands Locking Into Gesture technique: start with hands at different Y positions (but same Z) and expand or shrink distance in X and Y.

3.3 'Spindle with Raise' Technique

This technique is a modification of the Spindle technique [9]. A simple spindle technique (without translation) would only allow for y & z-axis rotation and scaling. Other contributions have modified this technique to also allow rotation around the x axis, such as the Spindle + Wheel technique, where rotation around the x axis can be achieved with isotonic input devices. The HoloLens hand tracking we rely on does not have the possibility of tracking hand rotations, thus we developed "Spindle with Raise", allowing for 4DOF (x-, y-, z-axis rotation + scale). For this technique, pitch rotation was incorporated by the user raising or lowering both hands along the y axis, which was the most intuitive available non-conflicted hand motion for pitch, as determined by extensive qualitative pilot testing. Figure 5 (left) illustrates the pitch gesture. Note that the Spindle stays operative during this gesture (same as for scale), so any small position changes of the two hands relative to each other will result in slight yaw, roll, or scale changes. This is a wanted effect as this method was designed as a fully unseparated 4DOF technique.

3.4 'Arcball with Two-Handed Scale' Technique

This technique used a one-handed arcball [6, 16] technique allowing for 3DOF rotation. The arcball is designed as a bounding sphere, fully enclosing the object to be manipulated, and represented as a fine wireframe mesh. A dot cursor (colored as described in 3) represents the 'grab point' on the ball surface, and is controlled by the hand position. Scale was achieved, as in the previous two techniques, by raising both hands in the "L" position, pinching, and pulling hands apart to make the object larger, and together to make the object smaller. This technique is similar to Bill McCrary's first iteration attempt of bimanual techniques on the Microsoft HoloLens [10]. It also closely resembles a design choice concluded from the work of Schlattmann et al., where they hypothesized that the combination of a two-handed technique and a one-handed technique could be advantageous in certain settings, but citing the need for further research into beneficial combinations [14]. Figure 5 (middle) illustrates a rotation sequence using the arcball, and Fig. 5 (right) demonstrates the associated two-handed scale.

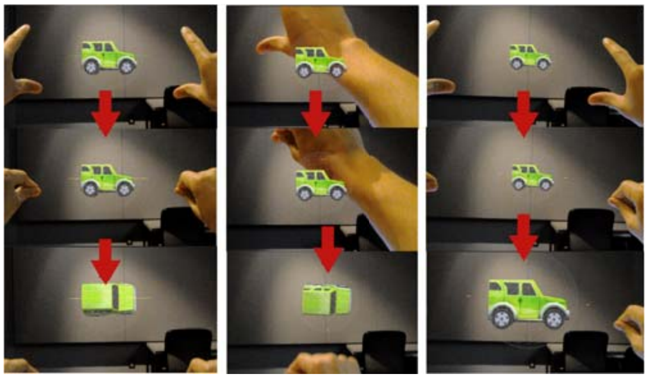

Figure 5: **Left**: Spindle with Raise example pitch rotation gesture sequence. **Middle**: One-Handed Arcball With Two-Handed Scale example pitch rotation gesture sequence. **Right**: Scale gesture sequence used with either technique. Red arrows simply indicate sequence and are not part of the user interface.

3.5 'Wireframe Cube' Technique

The Wireframe Cube technique is a modification of a one-handed object manipulation technique employed by standard programs shipping with the HoloLens.A wireframe bounding box is drawn around the object to be manipulated. Pinch points (corners and nodes) along the surface of the cube can be hovered over with head gaze and are highlighted in response, and overlaid with arrow indicators for the associated action. This action (axis rotation or scale) is then triggered with a finger pinch and drag, either left-right or top-down, both work for all actions. The current Wireframe cube technique on the HoloLens does not have the capability for pitch or roll rotations, thus it was modified to add x and z rotation pinch points (nodes) to allow it a fair comparison (see Fig. 6). We added two nodes each on the top and bottom plane of the wireframe box, all in the center (according to the original axis-aligned pose). Having fewer than the possible four edge nodes on the top and bottom simplifies the mapping of (now 2+4+2=8) nodes to the three cardinal axis rotations

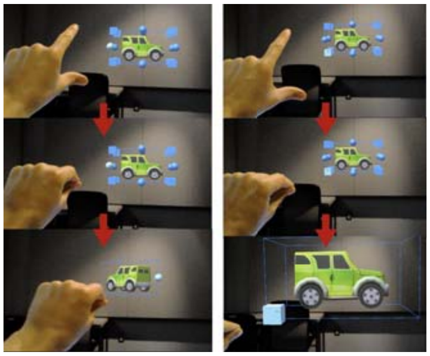

Figure 6: Wireframe Cube example yaw rotation gesture sequence (left) and scale gesture sequence (right). Red arrows simply indicate sequence and are not part of the user interface.

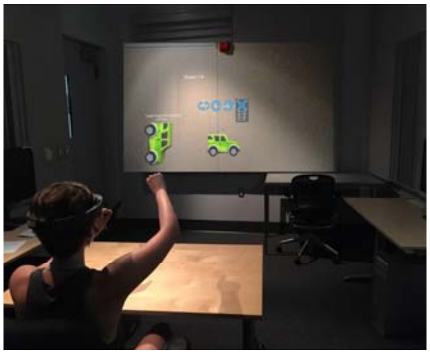

Figure 7: The room and setup of the user study (holograms added for illustration)

and also disambiguates the orientation of the bounding box on quick glimpse in case the object's rotation is not easily interpreted from its shape.

We decided to use the four middle nodes for yaw control, as in the HoloLens standard technique, and to dedicate the two top nodes to pitch control and the two bottom ones to roll.

4 USER STUDY

To evaluate the gestures both subjectively and objectively, we held a within-subjects user study with 48 participants, 28 female, and 20 male. The age range was 18 to 33 years, with the average age 22. Fig. 7 illustrates the setup. Three of these participants didn't complete the entire study and their partial data was not used at all in our evaluations. For seven additional participants, we experienced data recording issues on the HoloLens, so that we couldn't use their quantitative performance data. However, they were unaware of any problems and completed the whole study, so that we could count their qualitative impressions gathered through questionnaires. The user study began with a pre-study questionnaire. After completing this, users proceeded to the HoloLens ”Learn Gestures” tutorial to familiarize themselves with hand tracking and hand pinch gestures.

The user then started the fully-automated user study application on the HoloLens. From the beginning, the user was given visual and auditory instructions detailing the study. The study cycled through five sections, one for each of our five gesture techniques, randomized for each user to reduce ordering effects. Each section consisted of two parts, Training and Testing. Following the Testing of a technique, the user filled out an online questionnaire commenting on their assessments of the technique they just performed. The instructions for the entirety of the study were given both visually and aurally. Throughout the entirety of the study, all user interactions were logged for quantitative analysis.

4.1 Training

The training portion of each section had six parts to it to get the user comfortable with the manipulation technique.

Training: Part 1. The first part of training gave overall instructions on how to perform the manipulation technique along with a video looping through the different scale and rotation gestures. After listening to the instructions, the user could continue on to learn the scale portion of the gesture (Fig. 8, Left).

Training: Parts 2 - 5. These were for the scale, pitch, yaw, and roll portions of the manipulation technique training. Each stage involved instructions on how to perform the particular transformation action with the current manipulation technique, as well as a looping video demonstrating the appropriate way to perform the described action. The user was also given an augmented object and was told to practice the particular portion of the technique on the object until they felt comfortable with it. The user was not able to move on to the next training part until they performed the gesture on the object at least once (Fig. 8, Middle).

Training: Part 6. The final portion of training instructed users to practice all rotation and scale gestures until fully comfortable with them, and prompted the user that the following stage would be the testing portion of the technique.

4.2 Testing

This portion of each technique's section involved six rounds for quantitative user evaluation. Rounds 4-6 posed the very same test tasks as rounds 1-3, but in different order (avoiding a back-to-back repeat of the same task). Each round had an object to be manipulated on the right, as well as an object in a target pose (orientation and size) on the left (Fig. 8, Right). The user was instructed to attempt to scale and rotate the object on the right to match the orientation and size of the reference object on the left. We always used the same object: the green car depicted in Figures 5 through 8. The rotation/scale tasks all required rotation that was able to be resolved via just one single-axis rotation of 90deg, sometimes with an additional difference in scale, sometimes not. No translations were ever involved, and scale was always applied around the object center point, so that scaling never interfered with the rest of the transformations.

Even with these simplifications (which as a side-effect benefited separated-DoF techniques, see discussion in Section 6) participants commented about the difficulty of matching rotations. We arrived at this compromise setup through many iterations of extended pilot testing.

The user was also repeatedly told to complete each task as quickly and accurately as possible. Upon finishing each task, the user could select a button above the figure indicating they wanted to lock in the result and complete this round of testing. They were then prompted to rest their arms before proceeding to the next testing round. Upon final completion of the six rounds of testing for a technique, the user was given a questionnaire to record their qualitative impressions on the technique they performed.

5 RESULTS

In the following tables detailing our results, Hands Locking into Gesture, Hands Starting Positions, Spindle With Raise, Arcball, and Wireframe Cube are abbreviated as HLIG, HSP, SR, A, and WC, respectively. After a brief look at data collected during our training phases, we will report on Speed, Accuracy, and Qualitative Feedback results.

5.1 Training Results

Since participants had free reign as to how long they practiced each technique beyond some minimal requirements, we will take a quick look at the amounts of training time spent on the different techniques.

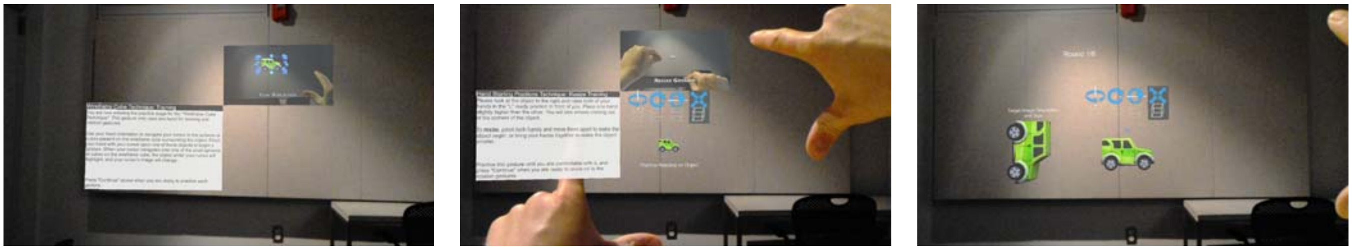

Figure 8: Training phases for user study. **Left:** Part 1, Overall Instructions. **Middle:** Example from Parts 2-6, training for scale, pitch, yaw, and roll, here: scale. **Right:** Example from Testing Rounds, here: Round 1 of 6

We also look at training times as a function of time into the entire training module, which reveals, not unexpectedly, that training sessions later in the training module tended to be shorter. That didn't disadvantage any particular technique since we statistically varied the order of techniques.

5.1.1 Training Results: Training Time per Technique

We compared mean completion time per section and performed a single-factor ANOVA, along with post-hoc Bonferroni-corrected pairwise T-tests to determine whether training time for participants varied significantly depending on the technique used. We found that Hands Starting Positions had a significantly longer training duration than that of Hands Locking into Gesture, Spindle With Raise, Arcball, and Wireframe cube (with Bonferroni corrected p-values of 4.87×10^{-6}, 0.0078, 3.6×10^{-7}, and 451×10^{-5}, respectively, see Fig. 9). Furthermore, Arcball had significantly lower training times than Spindle With Raise, which we speculate is due to people's familiarity with arcball rotations in 2D spaces. Between Hands Locking into Gesture, Arcball, and Wireframe cube, however, there was not a significant difference in training time, leading us to believe that although two-handed gestures are most likely less familiar to users than their one-handed counterparts, they do not necessarily require significantly more training.

5.1.2 Training Results: Training Time per Section

We also compared mean completion time per section and performed single-factor ANOVA, along with post-hoc Bonferroni-corrected pairwise T-tests to determine whether training time differed significantly throughout sections 1 - 5. We found that section 1 had significantly longer training times than sections 2 - 5 (Bonferroni corrected p-values of 2.28×10^{-7}, 3.12×10^{-8}, 8.61×10^{-12}, 4.38×10^{-19}, respectively). The same trend was largely present in subsequent sections, with training time for section 2 being significantly longer than that of section 4 and section 5 (Bonferroni corrected p-values of 0.0054 and 3.16×10^{-11}, respectively), and training time in section 3 and section 4 being significantly longer than that of section 5 (Bonferroni corrected p-values of 0.001 and 0.0102, respectively). A clear result from this analysis is that section 5 training time differed significantly from all sections preceding it, either indicating that users were impatient or fatigued by the end of the study, or that users were more familiar with gestures in AR and found training easier to complete and less need for it. We believe the latter to be true, and that our results were not significantly affected by this, as rotation accuracy did not significantly change from section 1 to section 5.

5.2 Timing Results

We compared mean completion time per gesture technique and performed a single-factor ANOVA, along with post-hoc Bonferroni-corrected pairwise T-tests to determine whether task time was influenced significantly by the technique used (Fig. 11). We found that Hands Locking into Gesture and Wireframe Cube both were significantly faster than Hands Starting Positions (p=0.00036 and p=0.00295 Bonferroni-corrected p-values respectively) and both were also significantly faster than Arcball ($p=4.05 \times 10^{-7}$ and $p=2.73 \times 10^{-8}$). Lastly, Spindle With Raise outperformed Arcball in terms of timing ($p=2.69 \times 10^{-5}$). Among these three 'speed winners', however, our tests did not indicate a significant difference ranking one over the other (Table 1).

Another timing result we noted was that there was a clear learning effect from Rounds 1-3 to Rounds 4-6 (see (Fig. 11). Remember that these rounds contained exactly the same transformation challenges: Rounds 4-6 repeated previous challenges from 1-3 in different order as a sanity check. The learning effect was significant ($p<6.0 \times 10^{-5}$).

5.3 Accuracy Results

Again, we performed a single-factor ANOVA, along with Bonferroni-corrected T-tests to determine whether task accuracy differed significantly among the techniques. To determine the accuracy to which a user completed a task, we took the angle delta in degrees (calculated via difference of quaternions) between the target object pose and the user's achieved pose upon task completion (indicated by clicking the

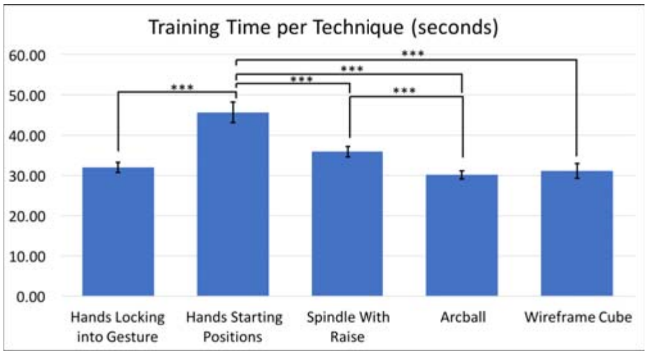

Figure 9: Training Time by Technique

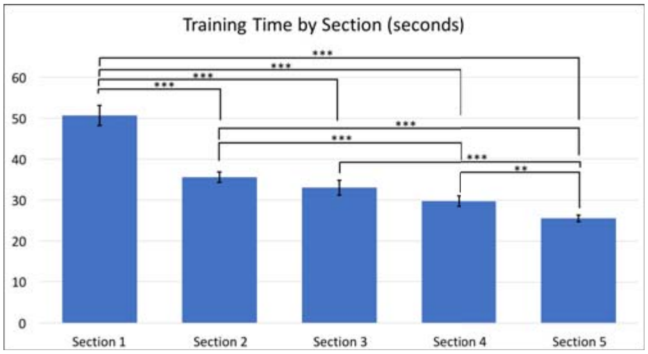

Figure 10: Training Time by Section

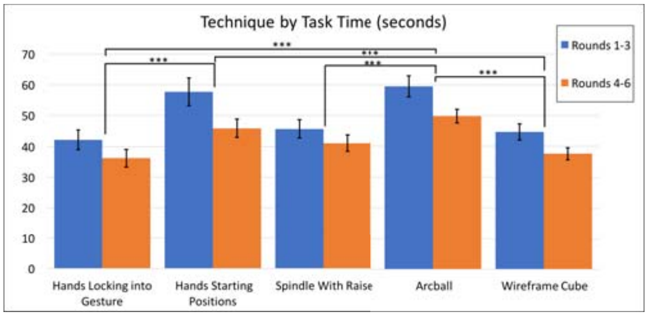

Figure 11: Technique by Task Time

Table 1: Timing Results: Pairwise Comparison of Techniques. Pairs with p-values > 0.1 not listed.

Techniques Compared	df	Bonferroni corrected p-value	alpha
HLIG - HSP	227	0.0003592132	0.05
HLIG - A	227	0.0000004045	0.05
HSP - SR	227	0.0579066759	0.05
HSP - WC	227	0.0029552669	0.05
SR - A	227	0.0000269066	0.05
A - WC	227	0.0000000273	0.05

complete button after each round). The maximum angle (indicating the worst a user could have done on the task) would be 180, thus we calculated accuracy as a percentage by taking $(180 - delta)/180$. We found that the Wireframe Cube outperformed Hands Starting Positions, Spindle With Raise, and Arcball in terms of rotation accuracy (Bonferroni corrected p-values of 8.24×10^{-5}, 0.0036, and 3.84×10^{-6}, respectively) Fig. 12. Between Wireframe Cube and Hands Locking into Gesture, we found no significant difference in rotation accuracy. We believe the larger margins of error for the other two-handed techniques may be due to user frustrations with the techniques due to lost hands or users not correctly remembering the gestures. Regarding scale accuracy, we found no significant differences between any techniques.

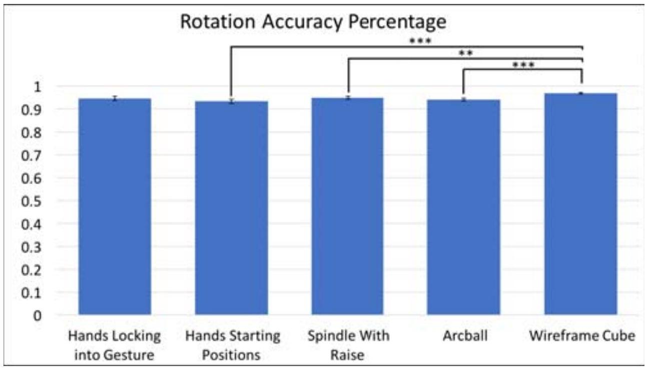

Figure 12: Rotation accuracy by technique with error bars showing standard error

5.4 Qualitative Feedback

Upon completion of the 6th testing round for each technique, the user was given a subjective survey in which they could rate on a Likert scale to what extent they felt several adjectives described the technique they had just been tested on. It turns out that the adjectives difficult/frustrating/tiring were closely correlated, as were the positive adjectives easy-to-perform/intuitive/enjoyable. We report here the results for 'frustrating' and 'enjoyable' as representatives for either group (see Fig. 13).

Table 2: Rotation Accuracy Results: Pairwise Comparison of Techniques. Pairs with p-values > 0.1 not listed.

Techniques compared	df	Bonferroni corrected p-value	alpha
HLIG - WC	227	0.0708903156	0.05
HSP - WC	227	0.0008237639	0.05
SR - WC	227	0.0360383770	0.05
A - WC	227	0.0000383814	0.05

We found that Hands Locking Into Gesture was significantly less frustrating to users than Hands Starting Positions and the One-handed Arcball (Bonferroni-corrected p-values of .00046 and .00012, respectively). Furthermore, Wireframe Cube was significantly less frustrating than Hands Starting Positions and Arcball (Bonferroni-corrected p-values of .00049 and .00059, respectively).

The pattern for enjoyment mirrors these findings, ie tasks which were less frustrating were enjoyed more by users. We found that users enjoyed Hands Locking Into Gesture over the one-handed Arcball technique, with a Bonferroni-corrected p-value of .00011.

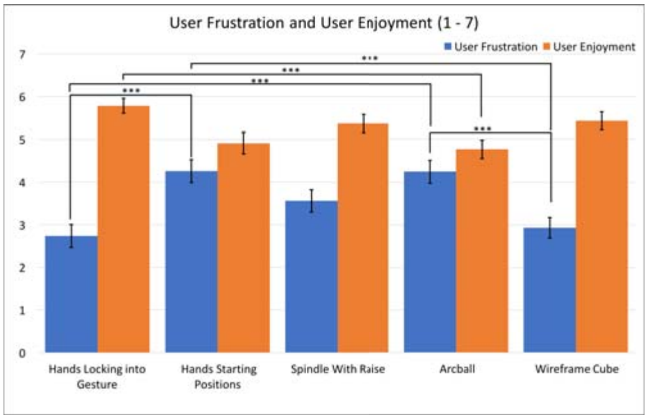

Figure 13: Qualitative Feedback: Frustration and Enjoyment

5.4.1 User Preference

Upon completion of the study, we gave users a subjective survey to determine which technique they preferred overall. Users were instructed to choose only one technique which was their overall preferred technique. We found that the Hands Locking Into Gesture technique gained overall user preference Fig. 14, with the Spindle and Wireframe Cube techniques competing for second place on aggregate.

6 DISCUSSION AND CONCLUSION

One setback for two-handed gestures on the HoloLens is increased hand-tracking losses. Fig. 15 shows the average number of hand losses and GS1 attempts made by participants for each technique. Future work on hand-tracking could work towards increasing the performance of hand tracking on the HoloLens to both lower user frustration with two-handed gestures and to increase their performance. A larger tracked interaction space in front of the user would limit these occurrences and likely improve the acceptance and performance of two-handed interaction techniques significantly.

Our results clearly indicate that bimanual gestures have a place in the future of mobile AR. Even on the current incarnation of the HoloLens, the 'Hands Lacking into Gesture' technique performed competitively with techniques mirroring current practice and state of the art, and it was qualitatively preferred by users in our evaluation.

Our results should not be taken as an indication of the superiority of DoF separation techniques over continuous techniques (such as

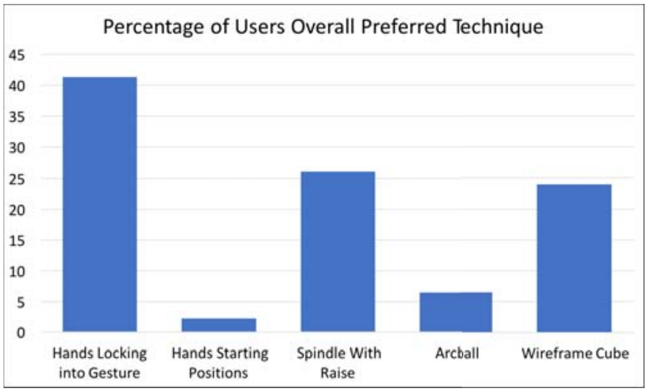

Figure 14: Preference of Technique

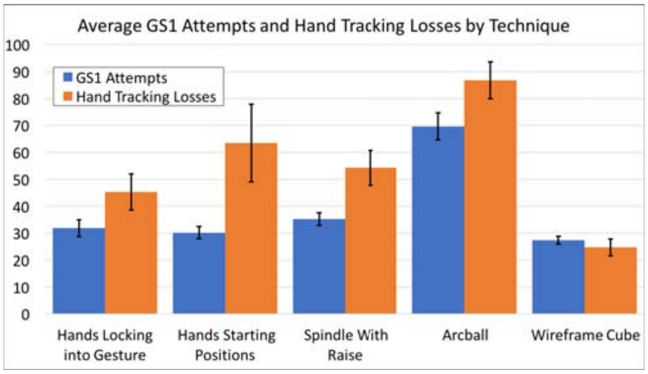

Figure 15: Average Hand Losses and Gesture Attempts

our Spindle With Raise). The simplification of our pose matching tasks (done for streamlining our experimental design) eliminated a strong disadvantage of DoF-separating rotation techniques: all rotation task we gave participants were solvable by a single cardinal-axis rotation of 90deg. We believe that more complex matching tasks would have boosted the performance of our continuous techniques (Spindle With Raise and Arcball). It is noteworthy that even with the current task framework the Spindle technique came in second in overall user preference (see 14).

Future work could limit the number of techniques to the Spindle With Raise technique and the Hands Locking Into Gesture technique for a closer look as what bimanual technique works best in what situation. Further integration of drawing annotations with scaling/rotating annotations could be integrated to explore which technique is preferred specifically with respect to manipulating annotations. Our results indicate that certain two-handed techniques perform comparatively to one-handed techniques in terms of accuracy and time, and in one instance gain the majority of user preference, showing that an environment for two-handed interactions on the HoloLens is justified and feasible.

ACKNOWLEDGMENTS

The authors wish to thank Adam Ibrahim and Yun Suk Chang for discussions and inspiration. This work was supported in part by ONR grant N00014-16-1-3002.

REFERENCES

[1] Y. S. Chang, B. Nuernberger, B. Luan, and T. Höllerer. Evaluating gesture-based augmented reality annotation. In *2017 IEEE Symposium on 3D User Interfaces (3DUI)*, pp. 182–185, March 2017. doi: 10.1109/3DUI.2017.7893337

[2] I. Cho and Z. Wartell. Evaluation of a bimanual simultaneous 7dof interaction technique in virtual environments. In *2015 IEEE Symposium on 3D User Interfaces (3DUI)*, pp. 133–136, March 2015. doi: 10.1109/3DUI.2015.7131738

[3] J. Feng, I. Cho, and Z. Wartell. Comparison of Device-Based, One and Two-Handed 7DOF Manipulation Techniques. In *Proceedings of the 3rd ACM Symposium on Spatial User Interaction - SUI '15*, pp. 2–9, 2015. doi: 10.1145/2788940.2788942

[4] D. Kline. Two Hands Gesture – Windows Mixed Reality Developer Forum, 2016. https://forums.hololens.com/discussion/1613/two-hands-gesture.

[5] R. Lala. *Quintilian A Framework for Intuitive Interaction in Immersive Environments*. PhD thesis, Media Arts and Technology, UCSB, 2011.

[6] J. J. LaViola, E. Kruijff, R. P. McMahan, D. Bowman, and I. P. Poupyrev. *3D user interfaces : theory and practice*. Addison-Wesley, second ed., 2017.

[7] T. Lee and T. Höllerer. Handy AR: Markerless Inspection of Augmented Reality Objects Using Fingertip Tracking. In *2007 11th IEEE International Symposium on Wearable Computers*, pp. 1–8. IEEE, oct 2007. doi: 10.1109/ISWC.2007.4373785

[8] J. C. Lvesque, D. Laurendeau, and M. Mokhtari. Bimanual gestural interface for virtual environments. In *2011 IEEE Virtual Reality Conference*, pp. 223–224, March 2011. doi: 10.1109/VR.2011.5759479

[9] D. Mapes and J. Moshell. A Two-Handed Interface for Object Manipulation in Virtual Environments. *Presence: Teleoperators and Virtual Environments*, 4(4):403–416, Jan. 1995. doi: 10.1162/pres.1995.4.4.403

[10] B. McCrary. HoloToolkit – Simple Drag/Resize/Rotate, 2017. https://www.billmccrary.com/holotoolkit-simple-dragresizerotate/.

[11] D. Mendes, F. Relvas, A. Ferreira, and J. Jorge. The benefits of dof separation in mid-air 3d object manipulation. In *Proceedings of the 22Nd ACM Conference on Virtual Reality Software and Technology*, VRST '16, pp. 261–268. ACM, New York, NY, USA, 2016. doi: 10.1145/2993369.2993396

[12] R. Owen, G. Kurtenbach, G. Fitzmaurice, T. Baudel, and B. Buxton. When it gets more difficult, use both hands: Exploring bimanual curve manipulation. In *Proceedings of Graphics Interface 2005*, GI '05, pp. 17–24, 2005.

[13] J. Raskin. Meanings, modes, monotony, and myths. In *The Humane Interface: New Directions for Designing Interactive Systems*, chap. 3. Addison-Wesley, New York, NY, USA, 2000.

[14] M. Schlattmann and R. Klein. Efficient bimanual symmetric 3d manipulation for bare-handed interaction. *Journal of Virtual Reality and Broadcasting*, 8, July 2010.

[15] U. Schultheis, J. Jerald, F. Toledo, A. Yoganandan, and P. Mlyniec. Comparison of a two-handed interface to a wand interface and a mouse interface for fundamental 3d tasks. In *2012 IEEE Symposium on 3D User Interfaces (3DUI)*, pp. 117–124, March 2012. doi: 10.1109/3DUI.2012.6184195

[16] K. Shoemake. Arcball: A user interface for specifying three-dimensional orientation using a mouse. In *Proceedings of the Conference on Graphics Interface '92*, pp. 151–156. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1992.

[17] P. Song, W. B. Goh, W. Hutama, C.-W. Fu, and X. Liu. A handle bar metaphor for virtual object manipulation with mid-air interaction. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '12, pp. 1297–1306, 2012. doi: 10.1145/2207676.2208585

[18] B. H. Thomas and W. Piekarski. Glove Based User Interaction Techniques for Augmented Reality in an Outdoor Environment. *Virtual Reality: Research, Development, and Applications*, 6(3):167–180, 2002.

[19] R. C. Zeleznik, A. S. Forsberg, and P. S. Strauss. Two pointer input for 3d interaction. In *Proceedings of the 1997 Symposium on Interactive 3D Graphics*, I3D '97, pp. 115–120. ACM, New York, NY, USA, 1997. doi: 10.1145/253284.253316