

**HUMAN POINT-TO-POINT REACHING AND
SWARM-TEAMING PERFORMANCE IN
MIXED REALITY**

by

CHEN ZHAO

Submitted in partial fulfillment of the requirements for the degree of
Master of Science

Department of Computer and Data Sciences

CASE WESTERN RESERVE UNIVERSITY

January, 2021

CASE WESTERN RESERVE UNIVERSITY
SCHOOL OF GRADUATE STUDIES

We hereby approve the thesis/dissertation of

Chen Zhao

candidate for the degree of **Master of Science***

Committee Chair

Michael Fu

Committee Member

Andy Podgurski

Committee Member

An Wang

Date of Defense

11/13/2020

*We also certify that written approval has been obtained
for any proprietary material contained therein

Contents

List of Tables	4
Abbreviations	9
Acknowledgments	10
Abstract	11
1 Introduction	12
1.1 3D Point-to-point reaching Task in MR and VR	12
1.2 Human-Swarm Teaming in Mixed Reality	13
1.3 Outline	14
2 Evaluation of Point-to-point Reaching Performance in Mixed Reality and Virtual Reality	15
2.1 Background	15
2.1.1 Fitts' Law	16
2.1.2 Virtual Reality	19
2.1.3 Mixed Reality	20
2.1.4 Mixed Reality and Virtual Reality in Rehabilitation	22
2.2 Methods	23
2.2.1 Experiment Paradigms	24

2.2.2	Equipment	29
2.2.3	Software Design	32
2.2.4	Internal Review Board protocol	35
2.2.5	Participants	36
2.3	Study Objectives	36
2.4	Performance Measures for Analysis	37
2.4.1	Throughput	37
2.4.2	End-point Error	37
2.4.3	Number of Corrective Movements	37
2.4.4	Efficiency	38
2.4.5	Initial Movement Error	38
2.4.6	Peak Velocity	38
2.4.7	Pre-Processing	39
2.4.8	Questionnaire	39
2.4.9	Statistical Analyses	40
2.5	Results	40
2.5.1	Throughput	40
2.5.2	End-point Error	42
2.5.3	Number of Corrective Movements	45
2.5.4	Efficiency	47
2.5.5	Initial Movement Error	49
2.5.6	Peak Velocity	51
2.5.7	Questionnaire	53
2.6	Discussion	56
2.6.1	Finger Pointing Task	58
2.6.2	Cup Placement Task	62
2.6.3	Trade-off between Reaching Motion Confidence and Accuracy	64

2.6.4	MR and VR Application Design Implications	64
2.7	Future Work	65
3	Human-Swarm Teaming Through Multi-user Interface With Mixed Reality	66
3.1	Background	67
3.1.1	Human-Swarm Teaming and Human-Swarm Interactions	67
3.1.2	Human Inputs for Human-Swarm Teaming	69
3.1.3	Mixed Reality’s capabilities for Human-Swarm Teaming	70
3.2	Methods	71
3.2.1	Hardware	71
3.2.2	Software Design	71
3.3	DARPA OFFSET Field Exercise 3	83
3.4	Experiment Procedure	83
3.5	Participant	85
3.6	Results and Discussion	85
3.7	Future Work	87
4	Conclusions	90
	Appendix A Questionnaire	92
	Bibliography	94

List of Tables

2.1	Target List for the finger pointing task	29
2.2	Target List for the cup placement task	29

List of Figures

2.1	An example of the Fitts' law experiment with target size W and distance to target D	17
2.2	The Virtuality Continuum diagram with some commercially devices.	21
2.3	The HoloLens 1 we used in our experiment	21
2.4	First person view of the finger pointing task for the condition where a fingertip cursor is visualized for the participant.	24
2.5	First person view of the condition where a virtual cup indicator is visualized for the participant.	25
2.6	Examples of depth perception problem.	26
2.7	Using HoloLens and shroud to simulate VR visualization	30
2.8	Physical workspace setup.	31
2.9	A system block diagram of components, communication contents and protocols.	32
2.10	A hollow cube with the exactly same size of Polhemus Liberty sensor for calibration.	34
2.11	Each participant's linear fit from movement time data and effective index of difficulty ID_e across all 60 trials	41
2.12	Histogram of R^2 values from all throughput linear regression showing the density of the distribution of the R^2	42
2.13	Bar plot with error bars for throughput measures of finger pointing tasks.	43

2.14	Each participant's linear fit from movement time data and effective index of difficulty ID_e across all 20 trials.	44
2.15	Histogram of R^2 values from all throughput linear regressions showing the density of the distribution of the R^2	44
2.16	Bar plot with error bars for throughput measures of cup placement tasks.	45
2.17	Bar plot with error bars for end-point error measures of finger pointing tasks.	46
2.18	Bar plot with error bars for end-point error measures of cup placement tasks.	47
2.19	Each participant's linear fit from number of corrective movement with corresponding target index of difficulty across all 60 trials.	48
2.20	Histogram of R^2 values from all number of corrective movements' linear regression which shows the density of the distribution of the R^2	48
2.21	Bar plot with error bars for # of corrective movement measures of finger pointing tasks.	49
2.22	Each participant's linear fit from number of corrective movement with corresponding target index of difficulty across all 20 trials.	50
2.23	Histogram of R^2 values from all number of corrective movements' linear regression which shows the density of the distribution of the R^2	50
2.24	Bar plot with error bars for # of corrective movements measures of cup placement tasks.	51
2.25	Bar plot with error bars for efficiency measures of finger pointing tasks.	52
2.26	Bar plot with error bars for efficiency measures of cup placement tasks.	53
2.27	Bar plot with error bars for initial movement error measures of finger pointing tasks.	54
2.28	Bar plot with error bars for initial movement error measures of cup placement tasks.	55

2.29	An example showed one participant's movement path for one of the cup placement task.	56
2.30	Each participant's linear fit from peak velocity with corresponding target index of difficulty across all 60 trials.	57
2.31	Histogram of R^2 values from all peak velocities' linear regression which shows the density of the distribution of the R^2	57
2.32	Bar plot with error bars for peak velocity measures of finger pointing tasks.	58
2.33	Each participant's linear fit from peak velocity with corresponding target index of difficulty across all 20 trials.	59
2.34	Histogram of R^2 values from all peak velocities' linear regression which shows the density of the distribution of the R^2	59
2.35	Bar plot with error bars for peak velocity measures of cup placement tasks.	60
2.36	Bar plots with error bars of six performance measure for finger pointing tasks and cup placement tasks.	61
3.1	Project concept.	67
3.2	Project's system block diagram	72
3.3	3D CWRU campus model in two different type. DARPA Distribution Statement "A" (Approved for Public Release, Distribution Unlimited).	73
3.4	EMG controls.	75
3.5	User select menu	76
3.6	Tactic planning simulation.	76
3.7	Tactic configuration process for SP and STR role	78
3.8	First person view of swarm actions and mixed reality environment for STF.	79
3.9	1:1 map overlaid on the physical environment.	79

3.10 A miniaturized map view of 5 drones implement a “Surveillance” tactic for STR.	81
3.11 A notification alert menu is waiting for STR’s decision on moving target.	81
3.12 Swarm hardware telemetry visualization at FX3.	84
3.13 Completion times of tactic planning process	86
3.14 Conceptual direct swarm control.	89

Abbreviations

HCI - Human-Computer Interaction

HMD - Head-mounted Display

VR - Virtual Reality

MR - Mixed Reality

VE - Virtual Environment

ANOVA - Analysis of Variance

HST - Human-Swarm Teaming

HSI - Human-Swarm Interaction

SP - Swarm Planner

STR - Swarm Tactician Rear

STF - Swarm Tactician Forward

Acknowledgments

Thank you, Dr. Michael J. Fu, for your patient guidance, encouragement, and the great opportunity of letting me being the part of our great lab. I could not done all these works without your help.

CWRU DARPA OFFSET Sprint 3 team, thank you guys for all the great teamwork and help on the project.

Thank you, my parents, for supporting me for so many years and still believing in me.

This work was accomplished with the generous support of Case Western Reserve University, the MetroHealth System, and the DARPA Contract No. N660011924023.

The views, opinions and/or findings expressed are those of the author and should not be interpreted as representing the official views or policies of the Department of Defense or the U.S. Government.

Human Point-to-point Reaching and Swarm-teaming Performance in Mixed Reality

Abstract

by

CHEN ZHAO

Immersive mixed reality (MR) and virtual reality (VR) displays enable use of hands to interact with virtual environments. Therefore, it becomes important to understand hand reaching performance VR/MR. The first part of this thesis evaluates human point-to-point reaching in VR/MR for finger and cup placement. Six measures studied for both tasks under different experiment conditions to determine what factors influence performance. A key finding was there is a trade-off between reaching motion confidence and accuracy. These findings support VR/MR interface design. The second part of this thesis focuses on the development of a multi-user MR interface that allows humans to collaborate with swarms of robotic drones. We demonstrated that this system can interface with a physical swarm test bed of 40 mobile and aerial robots and with a person with upper limb amputation.

Chapter 1

Introduction

In recent years, both Virtual Reality (VR) and Mixed Reality (MR) have not only been used in many fields: education, rehabilitation, and military, etc. but are also growing rapidly. Due to the difference between these two technologies, it is unknown how human performance differs between MR and VR. In this thesis, we focus on investigating Human performance in the context of i) three dimensional point-to-point reaching performance in MR and VR, ii) task performance of human-robotic-swarm teaming in MR.

1.1 3D Point-to-point reaching Task in MR and VR

Point-to-point reaching tasks have been used for evaluating human motor performance [1, 2]. Originally, the literature focused on models of point-to-point reaching tasks performed in one dimension, but later studies expanded knowledge of reaching for two and three dimensions [3, 4]. Since then, point-to-point reaching tasks have been used to evaluate human performance on many computer interface platforms.

MR and VR are both 3D computer-generated virtual environment (VE) display

modalities, and the newest technologies use the immersive head-mounted display for them that enable people to use their hands to interact with virtual objects. There is evidence that point-to-point reaching is affected by different VE display modalities [5, 6], but it is not known how human motor performance differs between immersive MR and VR head-mounted display modalities. Therefore, it is important for the design of virtual environment human interfaces by understanding how human reaching performance differs in VR/MR. Especially for applications that require speed or precision, such as robotic surgery, robot teleportation, or aviation drone interfaces, knowing the best virtual environment display modality is critical to safety and task performance.

The first aim of this thesis focuses on evaluating point-to-point reaching performance in various MR and VR conditions with both finger pointing and cup placement tasks. Experiment conditions include: MR environment with the virtual indicator of finger or cup, MR environment without the indicator, and VR environment simulated from modified MR device. We hypothesize that task performance under MR conditions will be better than VR condition for both finger pointing and cup placement tasks.

1.2 Human-Swarm Teaming in Mixed Reality

Human-swarm teaming (HST) involves concurrent interactions among humans and robot swarms. Swarms represent a group of distributed robots that can self-organize and generate group-level emergent behaviors from local communication [7].

There are many studies about using VR to communicate with swarm robots [8, 9], but the use of MR for human-swarm teaming is still innovative and novel. Immersive MR headset can provide tactical information related to the mission while preserving a person's view of the physical environment around them, which may benefit human-

swarm teaming. Also, the current literature has proposed methods for human-swarm teaming with swarms consisting of a limited number of robots, but it is not yet known how to best design human interfaces to allow humans teams with swarms consisting of hundreds of robots. The second aim of this thesis is to develop an MR human-swarm-interface to suit the operational needs of prototypical military personas and be used by people with and without below-elbow amputations.

1.3 Outline

In this thesis, Chapter 2 covers the evaluation of point-to-point reaching performance in MR and VR, which includes: a background of related previous studies, methods showing how the study was implemented, results of performance measures, discussion, and future work. Chapter 3 focuses on the implementation of an MR human-swarm teaming interface and has a similar structure: background of prior work, methods, results, discussion, and future work. Finally, this thesis concludes with the implications of these two aims in the field of virtual environment interaction.

Chapter 2

Evaluation of Point-to-point Reaching Performance in Mixed Reality and Virtual Reality

This chapter will evaluate the point-to-point reaching performance in immersive MR and VR head-mounted display modalities, as it is currently unknown which immersive virtual environment (VE) display modality is more beneficial for point-to-point reaching tasks. This chapter is in preparation for publication to the journal PRESENCE: Virtual and Augmented Reality [10].

2.1 Background

This chapter will focus on topics that are relevant to human motor performance (Fitts' law) in Mixed Reality (MR) and Virtual Reality (VR).

2.1.1 Fitts' Law

In 1954, Paul Fitts created a predictive model to determine the movement time for human point-to-point reaching motion in one-dimension, called Fitts' Law [1, 2]. In the original Fitts' task, the participants were asked to hold a stylus and tried to hit the designed target as quickly as possible. According to Fitts' law, an equation was proposed that the index of difficulty (ID) can be defined as:

$$ID = \log_2\left(2\frac{D}{W}\right), \quad (2.1)$$

where ID represents the difficulty level of the motor task, D is the distance between the start point and the center of the target, and W is the width of the target (Fig. 2.1). The units of ID are bits. Fitts also proposed that the average movement time (MT) should remain constant for different D and W as long as ID ratio is also constant,

$$MT = a + b \cdot ID, \quad (2.2)$$

where MT is the average movement time for a point-to-point movement, and the constants a and b are empirically determined by regression analysis of MT data for a given input device. The units of MT can be varied - if it is measured in seconds, the units of a are also seconds, and the unit of b is "bits/second".

In 1992, MacKenzie [11] proposed a Shannon formulation of Fitts law for HCI (Human-Computer Interaction). Soukoreff and Mackenzie also suggested that researchers should use the Shannon formulation of Fitts law since it can provide a better fit with observations. It is impossible to get a negative ID from this formulation [12]. The formulation of ID is:

$$ID = \log_2\left(1 + \frac{D}{W}\right), \quad (2.3)$$

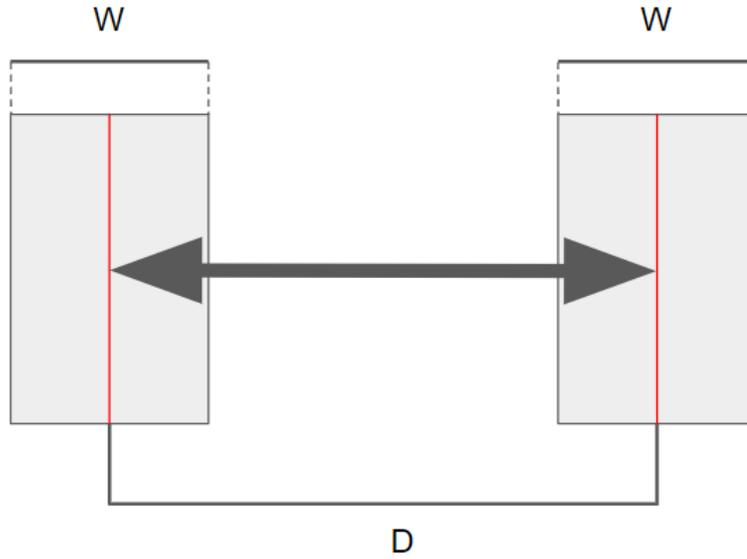


Figure 2.1: An example of the Fitts' law experiment with target size W and distance to target D

where ID is the index of difficulty, W is the width of a target, and D is the distance of a target from the start point.

Throughput

Originally, Fitts proposed an index of performance (IP) to measure human performance. IP is calculated by dividing the index of difficulty (ID) by the movement time (MT) of a reaching task,

$$IP = \frac{ID}{MT}. \quad (2.4)$$

IP has a unit of “bits/second”. Later in part 9 of the ISO 9241:2000 standard, Throughput (TP , in bits per second) has been brought up,

$$TP = \frac{ID}{MT}, \quad (2.5)$$

Due to human error, the end-point error will have a normal random distribution.

Therefore, Soukoreff and Mackenzie recommended using an effective index of difficulty (ID_e) instead of using ID [12].

$$ID_e = \log_2\left(1 + \frac{D_e}{W_e}\right), \quad (2.6)$$

where D_e is the effective distance which calculated as the mean movement distance between the start point and the target points, and the effective target width (W_e) defined as $W_e = 4.133\sigma$, where σ is the standard deviation. Therefore, to measure both speed and accuracy of TP , and to have only one dependent measure that can compare more than two experimental conditions, the new TP equation was defined as [12]

$$TP = \frac{1}{y} \sum_{i=1}^y \left(\frac{1}{x} \sum_{j=1}^x \frac{ID_{e_{ij}}}{MT_{ij}} \right), \quad (2.7)$$

where x is the number of movement conditions, and y is the number of participants.

Since the use of ID_e and TP can provide a better overall analysis of Fitts' task performance under multiple experimental conditions, we will be using these formulations to evaluate our experiment.

2D and 3D Extensions

Fitts' law was originally tested in one-dimension, but has been extended to two-dimensions and three-dimensions ever since [3, 4].

The 2D extension of Fitts' law has already been used to evaluate human-computer interfaces for years. Many applications and platforms benefit from it. One of the major applications for Fitts law in 2D is user interface design on 2D computer screens, which has become an essential part of our everyday life. The model of Fitts' law 2D extension is similar to in 1D but also adjusts to consider the geometry and errors of targets [3].

As for Fitts' law 3D extension, there also are many studies that show Fitts' law can

apply to 3D reaching tasks [4, 13, 14]. Murata and Iwase found that adding a third directional parameter into the traditional Fitts' law model can reduce the variance of the model [4]. Cha and Myung proposed another extended 3D Fitts' law model which includes the inclination angle and azimuth angle of a spherical coordinate system to the original Fitts' law formulation [14].

These extensions of Fitts law have been shown to have a better fit than the original model. Teather and Stuerzlinger studied 3D Fitts' tapping tasks based on ISO 9241-9, which used Shannon's formulation of Fitts' law and throughput. Their result showed closer targets are easier to hit than distant targets where a fish tank VR system was used [15]. It is hard to define task throughput using the above 3D extension models with many different parameters. For comparing experimental conditions, the current work uses Shannon's formulation of Fitts' Law and throughput as described in the previous sub-section.

2.1.2 Virtual Reality

Virtual Reality simulates a fully computer-generated digital environment that users can interact with. In the past years, Virtual Reality (VR) technology has been growing rapidly. Ivan Sutherland proposed the very first idea of VR in 1965 [16]. There are many platforms and methods for VR, such as fish tank VR [17], projector-based VR [18], and immersive VR head-mounted display [19]. However, the immersion of immersive VR head-mounted display is very persuasive because it tracks head and possibly hands movement to make interaction intuitive and natural [5, 20]. Also, VR head-mounted display allows users to walk around in a limited space to provide a more immersive experience. With the advantage of immersive VR head-mounted display and advancements in technology and hardware, more and more commercial VR headsets have started showing up since 2010. Common consumer VR headsets include Oculus Rift, HTC Vive, and Valve Index, etc. However, most of these VR

headsets required a wired connection with a computer. To free users from the constraint of wire, wireless adapter for VR (VIVE wireless adapter) and all-in-one VR headset (Oculus quest) were produced recently.

2.1.3 Mixed Reality

Mixed Reality (MR) overlays digital objects on real-world objects and allows interaction between them simultaneously. In 1994, Paul Migram and Fumio Kishino defined that MR is where the real world and virtual world objects are presented together within a single display [21]. The first fully immersive MR system was developed in 1992 as a Virtual Fixtures platform by Louis Rosenberg at the Armstrong Laboratories of the United States Air Force [22]. It showed that humans could control robots in a real-world environment with physical objects using a system that provided 3D computer-generated overlays. The study used Fitts' law to evaluate human performance in the environments with or without virtual objects as helpful information.

Many people may also recognize MR as Augmented Reality (AR) because they both add digital elements to the real-world environment. Also, the market circumstances are more focused on VR, so the distinction between MR and AR has not been clearly drawn yet. Paul Migram and Fumio Kishino proposed a way to visualize the differences between VR, AR, and MR called "virtuality continuum" [21]. The Virtuality Continuum (VC) shows the changes from a fully real to a completely virtual environment. Based on the original VC, We added some updated device and platform information to create a new VC graph (Fig. 2.2).

As the above Figure shows, MR is the combination of VR and AR. AR only overlies digital content onto the physical environment while MR does the same and allows interaction with both virtual and physical environments. Also, with the advent of Microsoft HoloLens version 1 (Microsoft Corp., Redmond WA) in 2016, MR has become more recognizable (Fig. 2.3). Later in 2018, another MR headset called

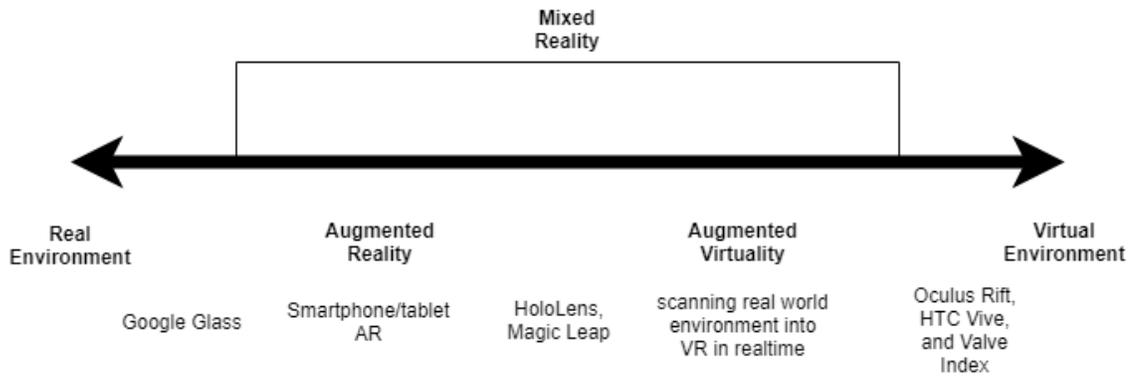


Figure 2.2: The Virtuality Continuum diagram with some commercially devices.

Magic Leap One (Magic Leap, Inc., Plantation, FL) got launched.



Figure 2.3: The HoloLens 1 we used in our experiment

Mixed Reality Head-mounted Display: HoloLens Version 1

Due to the nature of MR and the popularity of head-mounted display, almost all current commercial MR devices are head-mounted display, such as HoloLens and Magic Leap. Microsoft HoloLens version 1 was the first head-mounted display for

MR which runs the Windows Mixed Reality platform under Windows 10. HoloLens version 1 features several types of sensors, including: one inertial measurement unit (IMU), four environment understanding sensing cameras, one depth camera, one high definition (HD) video camera, MR capture, four microphones, and one ambient light sensor, all of which handle tasks such as spatial mapping, gesture recognition, and voice/speech recognition [23]. HoloLens version 1 also offers eye-based rendering where users can see computer-generated virtual objects by the reflection of two HD 16:9 light engines onto each eye. With the see-through holographic lenses, users can see both the real-world environments and virtual objects at the same time. Combining the functionality of sensors and display, users can actually interact with both real and virtual environments at the same time in realtime. HoloLens version 1 offers a limited field of view, which is about 30 degrees wide and 17 degrees high, and about 1268 by 720 pixels of resolution per eye. In 2019, Microsoft announced the new generation of HoloLens: HoloLens version 2 at the Mobile World Congress, which contains many improvements. One of those improvements HoloLens version 2 has a diagonal field of view of 52 degrees, compared to HoloLens version 1's 34 degrees. However, HoloLens version 2 was not released during our experiment, HoloLens version 1 is the MR device we used in our study.

2.1.4 Mixed Reality and Virtual Reality in Rehabilitation

Our study focuses on evaluating human motor performance in various MR and VR conditions for applying MR to rehabilitation. Rehabilitation facilitates motor learning, which leads to a relatively permanent change in motor behavior due to practice or experience [24]. Therefore, measuring human motor performance on different platforms (such as VR and MR) can help developing rehabilitation applications.

Over the past several years, VR technology has become an important part of the field of neuro-rehabilitation [25]. Based on VR's nature, users can interact with

virtual objects and navigate in a 3D computer-generated virtual environment (VE). It allows users to improve their motor skills and recover their muscles in an engaging, safe way. The advantages of VR-based rehabilitation over traditional therapies are the repetition of reaching tasks, feedback about motor performance, and motivation [26]. However, interaction with virtual objects is very different from physical objects, which have mass and volume.

MR allows users to interact with both real and virtual objects at the same time. By projecting computer-generated images onto the real world, MR technology can ease the transfer of rehabilitation exercises into everyday life [27]. MR environments can provide complex, adaptive scenes for interactive practice and feedback that engage the user physically and mentally [28]. However, it is not known if immersive MR display modalities affect point-to-point reaching. There is evidence that changes in VR display rotations and immersiveness can affect point-to-point reaching performance [5]. There is also a study that examined the point-to-point reaching performance with different virtual hand techniques in a tangible augmented reality environment [29]. However, there are no such studies have yet been conducted for comparing point-to-point reaching performance between immersive MR and VR modalities.

The purpose of the current study is to gain initial insight into these gaps in knowledge. We investigated the hypothesis that MR display modalities will benefit point-to-point reaching compared to VR modalities.

2.2 Methods

The following section explains the experiment of evaluating point-to-point reaching performance, equipment, software design, Institution’s Internal Review Board (IRB) protocol, participants, and experiment paradigms.

2.2.1 Experiment Paradigms

To evaluate point-to-point reaching performance, a variation of Fitts' law where uses ID_e instead of ID for throughput analysis was used [12], and a similar design for finger point-to-point reaching tasks were also borrowed for reference to implement this experiment [5].

We investigated two types of point-to-point reaching that are relevant to stroke rehabilitation: finger pointing (Fig. 2.4) and cup placement (Fig. 2.5). Point-to-point reaching is an important component of the computer and robot-assisted stroke rehabilitation. There is evidence that the virtual environment display modalities can significantly affect point-to-point reaching performance [5]. Therefore, it is important to understand better the effect that an immersive MR display modality has on these two types of point-to-point reaching.

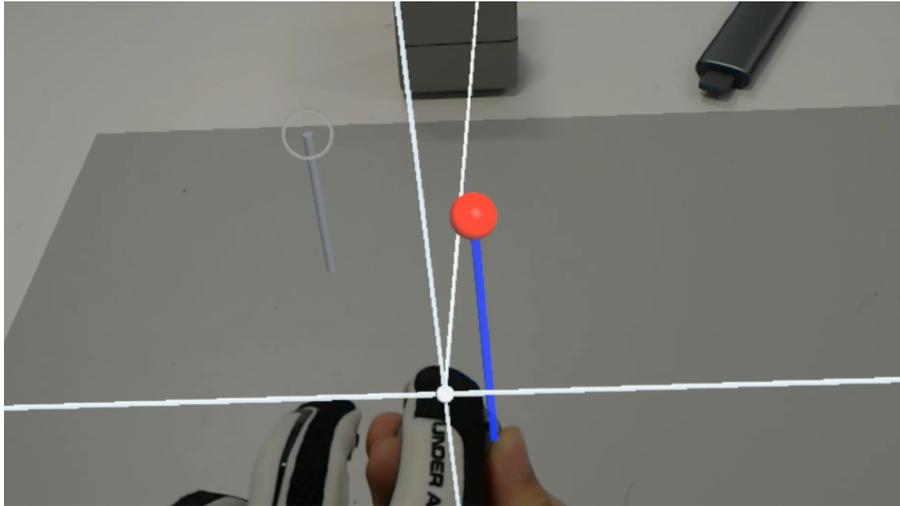


Figure 2.4: First person view of the finger pointing task for the condition where a fingertip cursor is visualized for the participant.

Finger pointing is relevant to stroke rehabilitation because it is required shoulder-elbow reaching, which is necessary to place the hand around a person's functional workspace in preparation for activities of daily living requiring the hand. Point-to-point reaching using a cup has not been studied in the context of MR, but it is also



Figure 2.5: First person view of the condition where a virtual cup indicator is visualized for the participant.

important to stroke rehabilitation. Cup placement is a fundamental activity of daily living that requires a person to grasp, relocate, and place a cup. Both finger pointing and cup placement are commonly trained during stroke rehabilitation.

Mixed reality displays have the potential to enable people with stroke to practice using physical objects to interact with virtual objects. Still, it is not known how best to design these interactions. Virtual stroke rehabilitation tasks have been developed that train the manipulation of virtual cups [30]. Virtual cup placement tasks allow people with stroke to practice repetitive hand use in a simulated environment with minimal consequences for failure. For instance, they do not need to worry about cleaning up spilled liquids. However, virtual cups lack mass and volume, so people with hand impairment will eventually need to practice grasping physical cups to improve their real-world function.

It may be beneficial to use a physical cup within a virtual environment so that users can continue to practice with minimal consequences for failure, but this has not yet been done in immersive MR. Others have developed stroke rehabilitation interventions using a cup-shaped color-marker to perform target reaching tasks in

both VR (computer monitor) and AR (projector) environment [31]. In the current experiment, we investigated immersive display modalities that enable a physical cup to interact with virtual targets directly.

We also investigated potential methods of indicating the location of the physical objects within the virtual environment. Depth perception can become impaired by immersive MR displays because typical depth occlusion cues are absent. Virtual objects do not become obscured by physical objects that are placed between it and the viewer (Fig. 2.6). Cursors and avatars linked to the physical objects can be used to provide depth cues (Fig. 2.4 and 2.5), but the impact of their use on point-to-point reaching has not yet been studied. The current study addresses this knowledge gap by investigating the effect that cursors have on immersive mixed reality reaching performance.



(a) Virtual target was over the physical cup. (b) Virtual target was over the index finger.

Figure 2.6: Examples of depth perception problem.

For both finger pointing and the cup placement task, Three conditions were tested: “MR with indicator” condition which participants perform point-to-point reaching tasks with a virtual cursor/cup indicator in MR environment (Fig. 2.4 and 2.5), “MR without indicator” condition that allow participants perform point-to-point reaching tasks without any virtual indicator in MR environment (Fig. 2.6), and “VR” condition that lets participants perform the same tasks with the virtual indicator

(cursor/cup) in a simulated VR environment.

For both tasks, participants were asked to sit in front of the physical workspace and wear the HoloLens. During the finger pointing task, participants were asked to wear a glove instrumented with an electromagnetic tracking sensor on their dominant hand that tracks their index finger position. As for the cup placement task, participants needed to hold a plastic cup that was instrumented with an electromagnetic tracking sensor attached under it. Then each participant was provided with about 5 minutes to practice before the actual experiments in both with or without virtual indicator conditions for both finger pointing and the cup placement task. Participants were also asked to perform all the reaching tasks as quickly and accurately as possible. For the finger pointing task, each of the three experiment conditions included a set of 60 point-to-point reaching trials (60 targets). For each trial, participants will observe a red sphere as the start position and a gray circle as the target (Fig. 2.4). Participants were instructed to put their index finger at the start position, then clicked a remote controller on their other hand when they were ready to move to the given target. This action changed the start position's and target's color to green and triggered audio feedback. Participants can click the remote controller again when they believed their fingertip reaches the center of the target. The start position's color would be changed to red, and the next generated target will be gray again. Another different audio feedback was provided for the second click. The cup placement task was very similar to the finger pointing task. There were only 20 point-to-point reaching trials (20 targets) for all three conditions of the cup placement task. The start position(a flat cylinder) and target (a hollow cylinder) of the cup placement task are both sat on top of the table with the same color and audio feedback pattern as the finger pointing task when participants performed cup placement tasks.

The start position was located near the edge of the table and vertically aligned with the electromagnetic sensor's source for both finger pointing and the cup placement

task. Due to the limited space of the physical workspace, the 60 targets of the finger pointing task were randomly selected from 15 unique targets spanning index of difficulty between 1.26 to 4.48 (Table 2.1), each repeated four times. Two of the repetitions were placed on the opposite side of the other two with respect to the start position to minimize the effect of directional bias. The 20 targets of the cup placement task were randomly constructed from 10 unique targets based on the index of difficulty between 1 to 2.44 (Table 2.2), each repeated two times similar to the finger pointing task. Even the movements were actually in 3D, but the workspace was in 2D. Therefore, it repeated two times horizontally for the cup placement task.

Table 2.1: Target List for the finger pointing task

ID	Distance (mm)	width(mm)
1.26	42	30
1.49	54	30
1.79	69	30
1.95	72	25
2.18	88	25
2.41	108	25
2.64	105	20
2.87	126	20
3.10	152	20
3.33	136	15
3.56	162	15
3.79	193	15
4.02	152	10
4.25	180	10
4.48	213	10

Table 2.2: Target List for the cup placement task

ID	Distance (mm)	width(mm)
1.00	70	70
1.16	86.5	70
1.32	97.5	65
1.48	116.5	65
1.64	127	60
1.80	149.5	60
1.96	159.5	55
2.12	184.5	55
2.28	193	50
2.44	221	50

2.2.2 Equipment

HoloLens was selected for this experiment since it is one of the most common immersive MR head-mounted display devices. We simulated an immersive VR head-mounted display by creating a custom hood to cover the HoloLens and obscure the physical environment (Fig. 2.7). HoloLens 1 only tracks hand position when it per-

forms a gesture, so we used a Polhemus Liberty electromagnetic sensor to provide constant, 6DOF tracking of hand position.

Initially, we attempted to use both HoloLens (MR) and Oculus Rift (VR) to evaluate point-to-point reaching performance between MR and VR. However, the Oculus Rift has a field of view of 110 degrees, which provided a very different experience than the 34 degree field of view of HoloLens. Therefore, to minimize any confounding effect due to the significant difference in the fields of view, a modification of HoloLens 1 was made to simulate the point-to-point reaching tasks in VR. As the figure shows(Fig. 2.7), we created a custom shroud to cover the HoloLens, so that no external objects could be seen by participants, which effectively converted HoloLens to a VR head-mounted display. For HoloLens to work properly, we use an Air Stick micro-suction tape to cover the front without blocking the sensors on top of the lens.



Figure 2.7: Using HoloLens and shroud to simulate VR visualization

We used the Polhemus Liberty to track the position of finger and cup that was used by participants for point-to-point reaching. It acquired 6 degrees of freedom (DOF) position and orientation tracking, at a rate of 240 HZ with a latency of 3.5 milliseconds. Since it is an electromagnetic tracker system, the equipment was set on a wood table without any metal near the physical workspace. The magnetic source of the sensor was set up in the middle of this physical workspace, and the virtual workspace was set in front of the source which also on top of the table. For the finger

pointing experiments, a Polhemus sensor attached to the index finger of a baseball batting glove (Under Armour Inc., Baltimore, MD) was worn on the participant's dominant hand. For the cup placement experiments, a Polhemus sensor was attached to the bottom of a plastic cup (Fig. 2.8).



Figure 2.8: Physical workspace setup. Cup and gloves on the left are for cup and finger experiments. The black bar-shape velcro on the right-bottom corner represents the location where participants need to be set in front of.

A Windows 10 pro Alienware 15 R3 laptop was used to run the integrating software to connect Polhemus Liberty and HoloLens. Data were sampled at 120 HZ from the Polhemus Liberty. A Logitech Spotlight Presentation Remote controller connected wirelessly via Bluetooth to the laptop was used by participants to click and confirm when they ready to reach and already reached the targets in each trial.

2.2.3 Software Design

The Unity3D game engine was used to develop the user interface for different tasks. C# was the programming language used in the implementation. To visualize the Polhemus Liberty sensor’s position and quaternion data on the HoloLens application, a server Unity project connected them. The server also transmitted human input from a Bluetooth remote controller to HoloLens, facilitated calibration, and recorded movements data (Fig. 2.9).

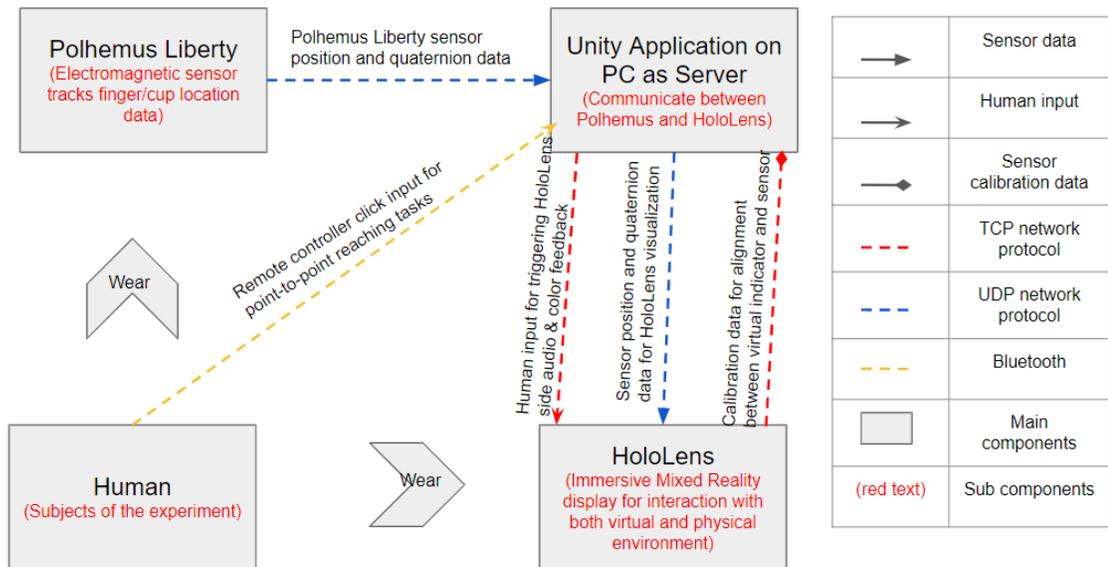


Figure 2.9: A system block diagram of components, communication contents and protocols.

Mixed Reality Calibration

HoloLens’ application treats the user’s head as the center of its own world coordinate. With the stabilized Polhemus Liberty magnetic source on top of the table and HoloLens initiating location, an approximated coordinate of the virtual workspace can be initially set up on the server that is then shared with the HoloLens application. In this way, the virtual workspace generated by HoloLens can roughly align

with the physical workspace and have a synchronized coordinate with the server’s virtual workspace. Users can also move and rotate the virtual workspace through keyboard control until it ideally sits on top of the workspace. At this point, the virtual workspace of the server and HoloLens became the connection point between two world coordinate systems. The next step is to synchronize the Polhemus Liberty sensor’s coordinate for both server and HoloLens. Since the sensor’s coordinate information on the server is also shared with the HoloLens application, the indicator that visually represents the sensor’s coordinate on HoloLens appears parallel to the sensor with certain offset. To visually align the Polhemus Liberty sensor with the virtual indicator on HoloLens, a virtual hollow cube with the same size as the real sensor that located on the near edge of the workspace was used(Fig. 2.10). Therefore, users can put the Polhemus Liberty sensor inside the virtual cube and press a key to calculate the offset of the position and quaternion between the physical sensor and the cube. This offset is shared and applied on both server and HoloLens sides, so the Polhemus Liberty sensor and virtual indicator can be aligned (Eq: 2.8 and 2.9). The offset data was recorded with an additional function to provide the ability for offset recalculation in case of misalignment by the operator. At this point, the calibration process has completed, users can choose to keep showing or hide the virtual indicator for different experiment conditions.

$$\text{Offset}_{\text{position}} = \text{Vector}_{\text{hollowCube}} - \text{Vector}_{\text{sensor}} \quad (2.8)$$

$$\text{Vector}_{\text{sensor}} = \text{Vector}_{\text{sensor}} + \text{Offset}_{\text{position}},$$

$$\text{Offset}_{\text{quaternion}} = \text{Quaternion}_{\text{sensor}}^{-1} \cdot \text{Quaternion}_{\text{hollowCube}} \quad (2.9)$$

$$\text{Quaternion}_{\text{sensor}} = \text{Quaternion}_{\text{sensor}} \cdot \text{Offset}_{\text{quaternion}},$$

To transmit Polhemus Liberty sensor data to the server, a third-party plug-in package called “PIStream” provided by Polhemus was used. It constantly sent the

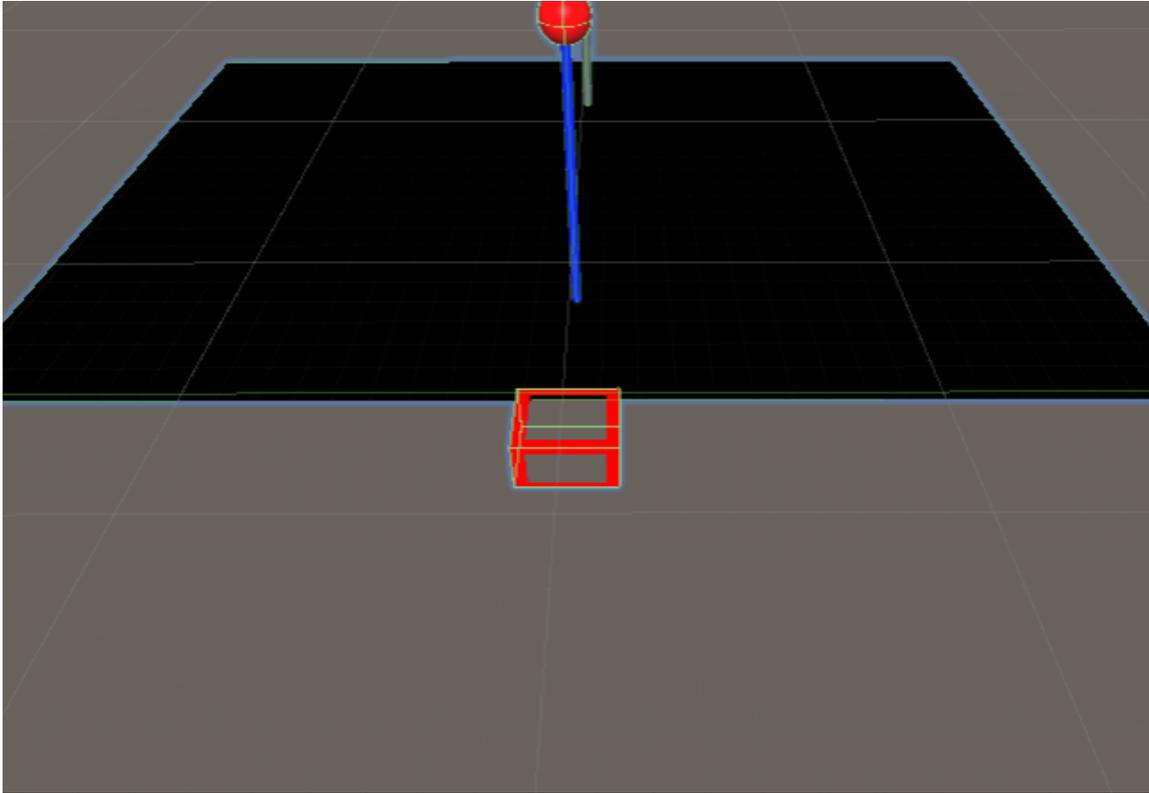


Figure 2.10: A hollow cube with the exactly same size of Polhemus Liberty sensor for calibration.

Polhemus Liberty sensor data through the User Datagram Protocol (UDP). After the server received the raw data, it would take the 6DOF data and convert them to Unity's left-handed, y-up world coordinate with a 1 meter Unity world unit that was also used in the HoloLens application. Additionally, the Polhemus Liberty sometimes reversed the sensor's orientation data (providing data showing that the sensor moved rightward when its vertical motion was leftward). Another function had to be developed to correct both position and rotation reversals which allow users to manually adjust the reversal errors based on the visual observation on the server application.

HoloLens Data Transfer

HoloLens is a standalone device that cannot connect to USB or Bluetooth peripherals, so the system was designed to be server-oriented. A different Unity software

for HoloLens was developed separately. Therefore, the data from the Polhemus Liberty sensor and human input can be sent to HoloLens through the server. Also, to record the movement data of the point-to-point reaching tasks effectively and easily, the recording process took place on the server side. For the convenience of data recording and because there are two independent Unity applications for the server and HoloLens (two different world coordinate systems), a server-based calibration process was required at the beginning of each trial.

Customized network connections were developed on both server and HoloLens side for transferring data. There are two types of data sending from the server to HoloLens. The first type of data is for HoloLens to visualize Polhemus Liberty sensor's virtual representation and virtual environment (virtual workspace and targets)'s position and quaternion data. Since the sensor data need to be continuously transmitted to HoloLens with less latency, the server would serialize the data into an Extensible Markup Language (XML) stream and send the message to HoloLens through UDP. The second type of data was sent from the server to HoloLens is human input from the Bluetooth remote controller and PC keyboard. The human input data included responses for the calibration process and initiating/terminating each movement of point-to-point reaching tasks. This type of data was also serialized and sent to HoloLens side through Transmission Control Protocol (TCP). Another type of data is sending calculated offset data between Polhemus Liberty sensor and virtual indicator from HoloLens to the server for synchronization. This data is also serialized and sent through TCP.

2.2.4 Internal Review Board protocol

The human experiment methods described below were reviewed by the Case Western Reserve University Institutional Review Board and approved as protocol number STUDY20181061.

2.2.5 Participants

Fifteen participants (ten male and five female, ages 19 - 32) were recruited and compensated with a \$10 cafe gift card for their participation in the study. Two participants were left-handed, others were all right-handed. Each participant was asked to perform six different point-to-point reaching tasks: three separate conditions for the finger pointing task, and three conditions for the cup placement task. The total time spent on the experiment per participant was about 2 hours. Unfortunately, due to the pandemic, we were only able to recruit 15 participants.

2.3 Study Objectives

This work studied how different VE environments (experiment conditions) affect human performance on 3D Fitts' task in the contexts of finger pointing and cup placement. There are 3 experiment conditions: MR environment with an indicator (virtual fingertip pointer or 3D cup model), MR environment without an indicator (pure interaction between physical and virtual object), and VR environment (shrouding the HoloLens to visually hide the physical environment, showing only virtual object interactions) with indicator displayed.

Objective 1: Compare the finger pointing task performance between MR and VR environments. Hypothesis 1: We hypothesize that finger pointing tasks have better performance under MR environments than VR environment.

Objective 2: Comparing the cup placement task performance between MR and VR environments. Hypothesis 2: We hypothesize that cup placement tasks have better performance under MR environments than VR environment.

2.4 Performance Measures for Analysis

The following six measurements were calculated based on the movement trajectory data from participants.

2.4.1 Throughput

The throughput measurement was described in detail in section 2.1.1. In this study, the task throughput represents is represented as $ID_e / (\text{movement time})$, which is also referred to as task completion rate. Since throughput and the target reaching times are inversely proportional across all target difficulties, a higher throughput value means better task performance.

2.4.2 End-point Error

End-point error is the Euclidean distance between the end point of a movement path and the target’s central location. All our participants are able-bodied, so lower the end-point error means better task performance.

2.4.3 Number of Corrective Movements

The number of corrective movements was defined as the number of direction changes across each movement path. Direction changes also can be determined by the local maxima of the acceleration signal. The smoother the movement path, the less the number of corrective movements. This analysis method was used in [32, 5] to measure human reaching performance in both virtual and real environments, with a larger number of corrective movements indicating worse task performance. In our case, we find the local maxima of acceleration using the derivative of velocity data and the “argrextrema” method of SciPy. The final value was calculated by adding the number of corrective movements on each axis and dividing by three (three axes).

Since the number of corrective movements can be correlated with target difficulty [5], we also consider the possible effect of ID on the number of corrective movements.

2.4.4 Efficiency

Efficiency was defined as how far the actual movement path deviated from the direct path to the target. The first formula of efficiency was defined in [33]. In our case, the formula is

$$\text{Efficiency} = \frac{D_{\text{endpoint}}}{D_{\text{path}} - D_{\text{endpoint}}}, \quad (2.10)$$

where D_{endpoint} is the euclidean distance from the initial movement point to the end-point of the reaching motion. D_{path} indicates the length of the path between initial and end-points. Higher efficiency means the movement path is more likely to be a straight line. Therefore, we assumed the increased efficiency means better task performance.

2.4.5 Initial Movement Error

Initial movement error is the magnitude of the difference between two normalized vectors: the initial movement vector and the target vector. The initial movement vector points from the initial movement position to the first corrective movement position. The target vector points from the initial movement position to the current target position. Since increased initial movement error may result in a longer reaching movement path, we assume larger initial movement error means worse task performance.

2.4.6 Peak Velocity

Peak velocity is the highest absolute value of velocity for each reaching path. Based on [32, 5], we can assume the higher peak velocity can be considered as motor con-

trol confidence and better task performance. To calculate the peak velocity, we will perform a root sum squared where we take the squared root of the sum of squares of the peak velocity on each axis.

2.4.7 Pre-Processing

Prior to analysis, trajectories from each trial were pre-processed to eliminate cases of remote controller click error and dwelling behavior that was caused by the participant keeping their finger or cup on the start or target positions without moving. To remove dwell time, only trajectories with a velocity higher than a threshold of 1.5mm/s were used in the analysis. The 1.5mm/s velocity threshold was chosen based on [32], where expert retinal surgeons' hand tremor frequency when they hold a stylus grip. To determine the effective data, we calculated the velocity based on the first derivative of the positional data. Also, to reduce noise, we performed a 5 Hz third-order lowpass Butterworth filter using the Python library SciPy. Also, eight outlying trials were removed that were affected by the remote controller click not being registered by the server. In these trials, the participant's trajectories included not only the path from the start position to the target, but also the extraneous return path from the target back to the start position because the remote controller press they entered at the target location was not registered.

2.4.8 Questionnaire

After the first two participants, we added a qualitative questionnaire to assess participants' user experience impressions of the MR device in case it became a factor in performance. The questionnaire was borrowed and modified based on Holger and Thomas's work [34]. This questionnaire contained 13 seven-point Likert scale questions of which seven questions are related to the participants' visual impression. Others are for previous experience and acceptance. We divided all participants into two

groups based on the mean value of these seven questions where “strongly disagree” is counted as one point and “strongly agree” is counted as seven points. Then we took a look at the six performance measures we mentioned above within these two groups. The questionnaire can be found in the appendix A.

2.4.9 Statistical Analyses

Hypothesis tests were performed for each performance measure to compare mean differences between the three experimental conditions using repeated measures analysis of variance (ANOVA) with Greenhouse-Geisser epsilon corrections, and Tukey’s honestly significant difference (HSD) multiple comparison post-hoc tests. All tests were performed using Python 3.6 with Pingouin package. Statistical power for each performance measure’s ANOVA test was computed to be 0.52 (calculated by G*Power 3.1 with Cohen’s medium effect size of 0.25, $\alpha = 0.05$, sample size of 15, one group, and three repeated measurements).

2.5 Results

Six performance measures were performed for each task. Since our study includes two tasks: the finger pointing task and the cup placement task, we will describe them separately.

2.5.1 Throughput

As described in section 2.4.1, the relation between throughput and target reaching time are inversely proportional. As evidence of this, ANOVA revealed that target difficulty was found to have a significant effect on movement time for both the finger pointing task ($p \leq 0.001$, $F(14, 196) = 28.66$) and the cup placement task ($p \leq 0.001$, $F(9, 126) = 7.93$) tasks. Therefore, reports of linear regression and histograms of

all R^2 values are shown for both tasks. Also, bar plots with error bars of these two tasks described the computed throughput values for different experiment conditions. The statistical tests for significant differences were calculated using throughput as the dependent variable and experiment conditions as the within-subjects factor.

For the finger pointing task, Figure 2.11 shows an example of the linear regression between 3 participants' movement time and ID_e . A histogram of linear regression R^2 for all participants is shown in Fig. 2.12. A bar plot with error bars of throughput across all experiment conditions are reported in Fig. 2.13). The effect of experiment condition on throughput for the finger pointing task was statistically significant ($p = 0.012$, $F(2, 28) = 5.17$). Tukey's HSD (Fig. 2.13) showed that the "MR without indicator" condition was found to have the highest mean throughput of (1.79 bit/s), which showed to be different from the "MR with indicator" (1.45 bit/s) and "VR" (1.47 bit/s) conditions. However, the mean differences were not statistically significant between the "MR with indicator" and "VR" conditions.

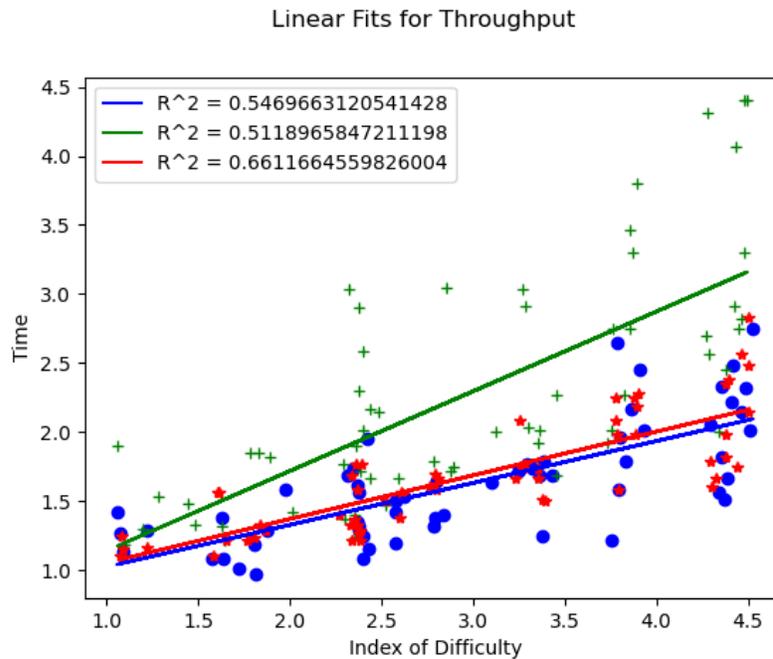


Figure 2.11: Each participant's linear fit from movement time data and effective index of difficulty ID_e across all 60 trials .

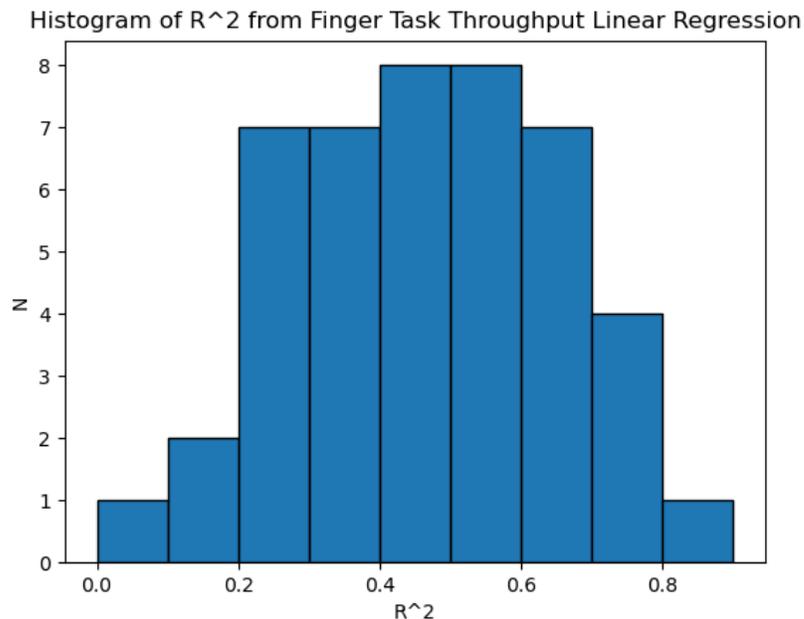


Figure 2.12: Histogram of R^2 values from all throughput linear regression showing the density of the distribution of the R^2 .

Similarly, for the cup placement task, 3 participants' linear fits for throughput (Fig. 2.14), histogram of all R^2 values (Fig. 2.15), and the bar plot with error bars for each condition (Fig. 2.16) are also reported. However, the experiment conditions had no significant effect on throughput ($p = 0.21$ $F(2, 28) = 1.66$).

2.5.2 End-point Error

Since no linear relationship was found between end-point error and target index of difficulty, we only calculated the end-point error results for each participant across all 3 experiment conditions. Statistical tests for significant mean differences were calculated with the within-subjects factor (experiment condition) and dependent variable (end-point error).

For the finger pointing task, the analysis of variance showed that the experiment conditions had a significant effect on end-point error ($p \leq 0.001$, $F(2, 28) = 30.75$). From Fig. 2.17 and Tukey's HSD multiple comparisons, experiment condition "MR

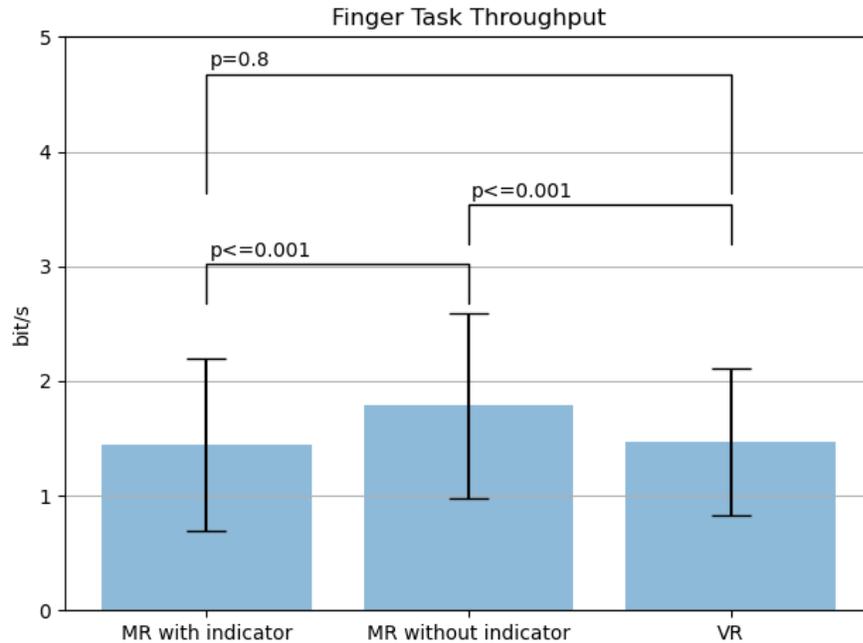


Figure 2.13: Bar plot with error bars for the finger pointing task throughput across all 3 experiment conditions. The mean throughput values for 3 conditions are: “MR with indicator” = 1.45 bits/s, “MR without indicator” = 1.79 bits/s, and “VR” = 1.47 bits/s. Higher throughput means better performance. “I”-shaped error bars stands for the standard deviation, which show the variability of the throughput value for each condition. Statistical significance differences between 3 conditions are also shown.

without indicator” had the highest end-point error mean value 0.013 m. “MR with indicator” condition was also higher than “VR” on end-point error mean values for the finger pointing task.

As for cup placement task, we also found experiment conditions had a significant effect on end-point error ($p = 0.004$, $F(2, 28) = 6.90$). Similar result to the finger pointing task, “MR without indicator” condition also had the highest end-point error mean value 0.01 m. The “MR with indicator” condition’s end-point error mean value 0.008 m was significantly higher than the “VR” condition (Fig. 2.18).

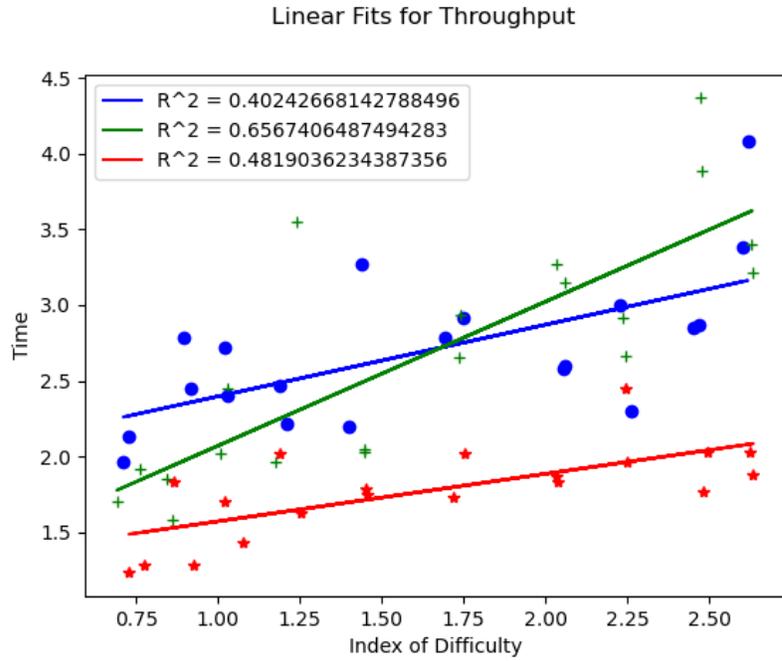


Figure 2.14: Each participant's linear fit from movement time data and effective index of difficulty ID_e across all 20 trials.

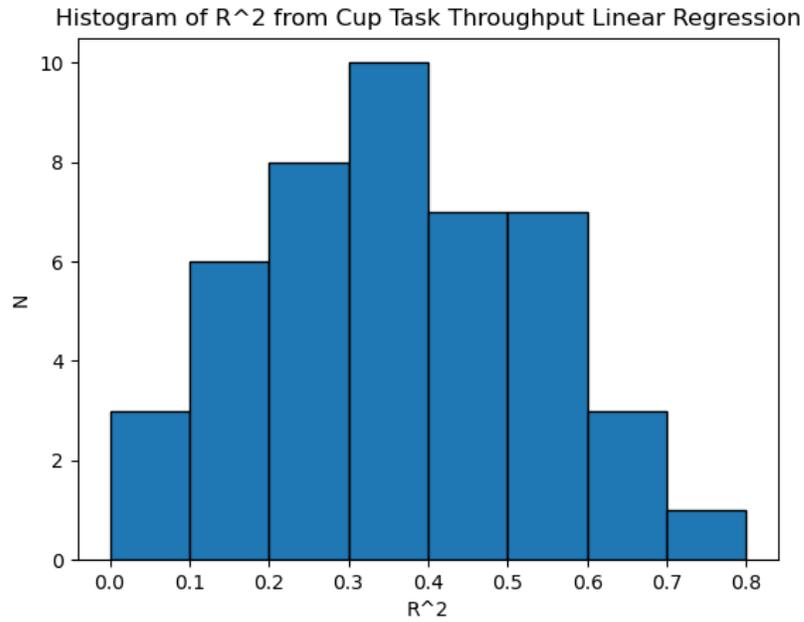


Figure 2.15: Histogram of R^2 values from all throughput linear regressions showing the density of the distribution of the R^2 .

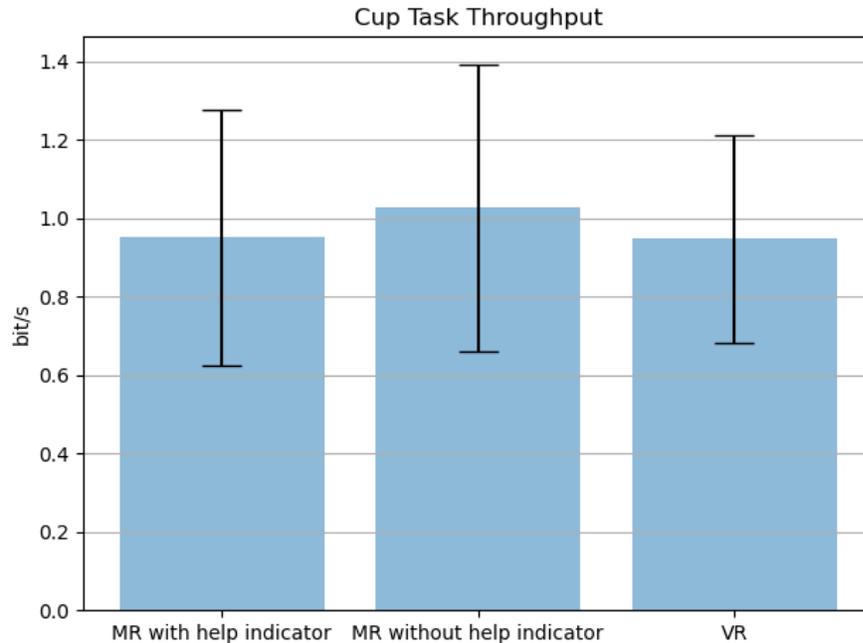


Figure 2.16: Bar plot with error bars for the cup placement task throughput across all 3 experiment conditions. The mean throughput values for 3 conditions are: “MR with indicator” = 0.95 bits/s, “MR without indicator” = 1.03 bits/s, and “VR” = 0.95 bits/s. Higher throughput means better performance. “I” shaped error bars stands for the standard deviation, which shows the variability of the throughput value for each condition.

2.5.3 Number of Corrective Movements

ID had a significant effect on the number of corrective movements for both the finger pointing task ($p \leq 0.001$, $F(14, 196) = 14.75$) and the cup placement task ($p \leq 0.001$, $F(9, 126) = 4.864$). We reported examples of linear regression from three participants and R^2 values from all participants’ linear regression as plots. Bar plots with error bars are also included for both finger pointing and the cup placement task. Statistical tests for significant mean difference were calculated with the experiment condition as the within-subjects factor, and the number of corrective movements results as the dependent variable.

For the finger pointing task, Fig. 2.19 reports three participants’ linear regression slopes from the number of corrective movements and target difficulty. A histogram

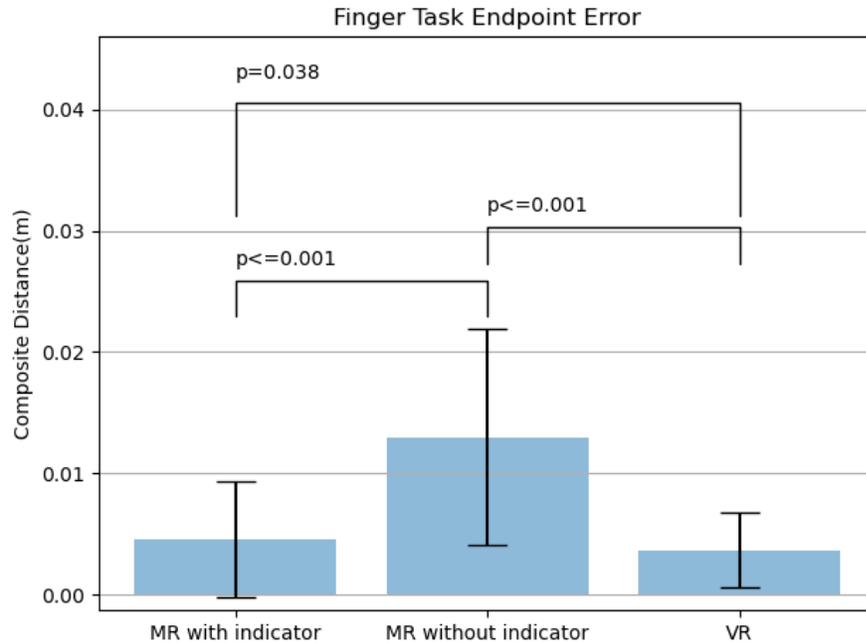


Figure 2.17: Bar plot with error bars for the finger pointing task end-point error across all 3 experiment conditions. The mean end-point error values for 3 conditions are: “MR with indicator” = 0.005 m, “MR without indicator” = 0.013 m, and “VR” = 0.004 m. Lower end-point error means better performance. “I” shaped error bars stands for the standard deviation, which shows the variability of the end-point error for each condition. Statistical significance differences between 3 conditions are also shown.

of R^2 values from all linear regression are plotted in Fig. 2.20. From statistical tests, experiment conditions had a significant effect on number of corrective movements ($p = 0.015$, $F(2, 28) = 4.87$). Fig. 2.21 and multiple comparisons showed that the “MR without indicator” condition had a significantly lower mean value of the number of corrective movements (2.21) than other conditions. Also, the “VR” condition’s mean value 2.62 is lower than “MR with indicator” conditions’ 3.21.

As for the cup placement task, the same types of plots to the finger pointing task are showed: 3 examples from linear regression (Fig. 2.22), a histogram of all linear regressions’ R^2 values (Fig. 2.23), and a bar plot with error bars for the number of corrective movements results (Fig. 2.24). However, experiment conditions did not have a significant effect on the number of corrective movements ($p\text{-GG-corr} = 0.33$

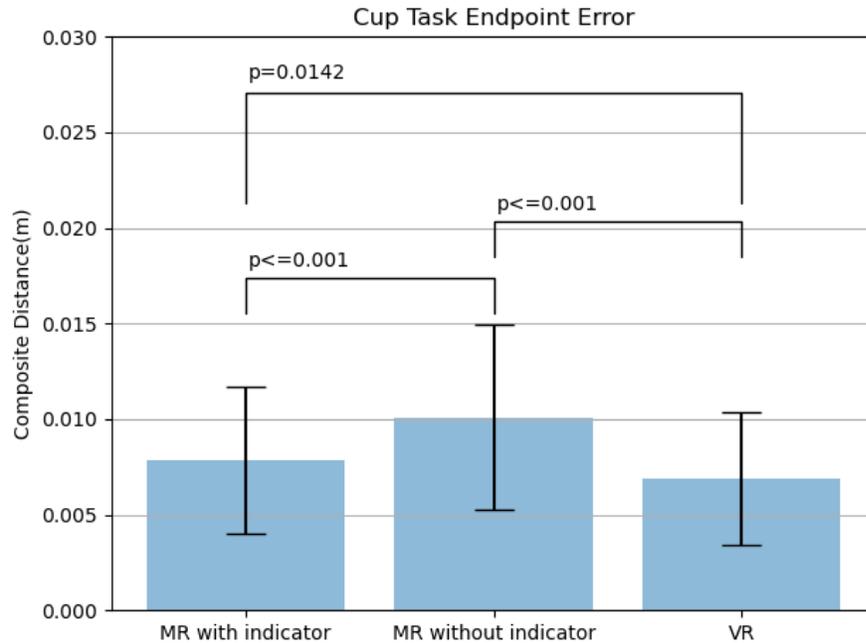


Figure 2.18: Bar plot with error bars for the finger pointing task end-point error across all 3 experiment conditions. The mean end-point error values for 3 conditions are: “MR with indicator” = 0.008 m, “MR without indicator” = 0.010 m, and “VR” = 0.007 m. Lower end-point error means better performance. “I” shaped error bars stands for the standard deviation, which shows the variability of the end-point error for each condition. Statistical significance differences between 3 conditions are also shown.

$F(2, 28) = 1.088$.

2.5.4 Efficiency

Efficiency was calculated for each participant across every experiment condition without considering the targets’ difficulties. Statistical tests for significant mean differences were computed with the dependent variable (efficiency) and within-subjects factor (experiment conditions).

Both bar plots with error bars for the finger pointing task (Fig. 2.25) and the cup placement task (Fig. 2.26) are showed below. However, experiment conditions were found that had no significant effect on efficiency for both finger pointing ($p = 0.23$, $F(2, 28) = 1.296$) and cup placement ($p = 0.12$, $F(2, 28) = 2.199$) tasks.

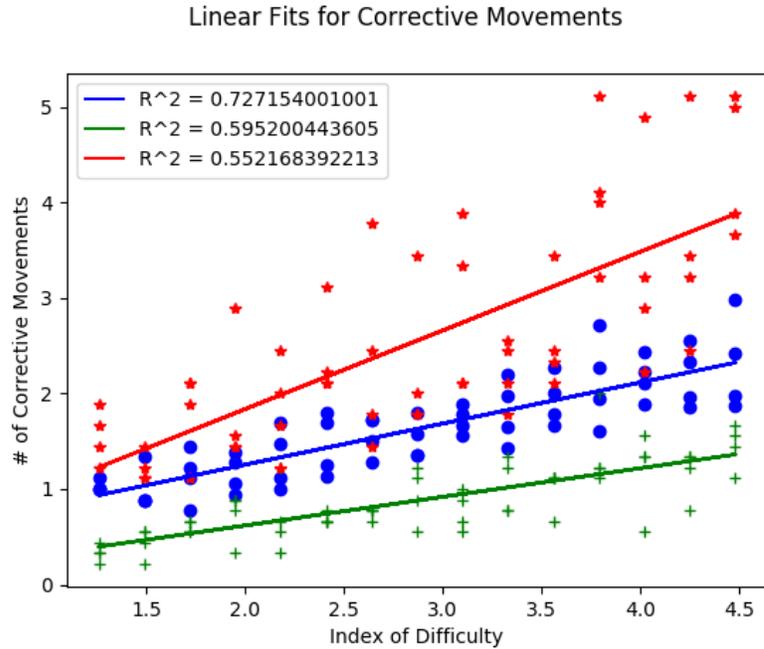


Figure 2.19: Each participant's linear fit from number of corrective movement with corresponding target index of difficulty across all 60 trials.

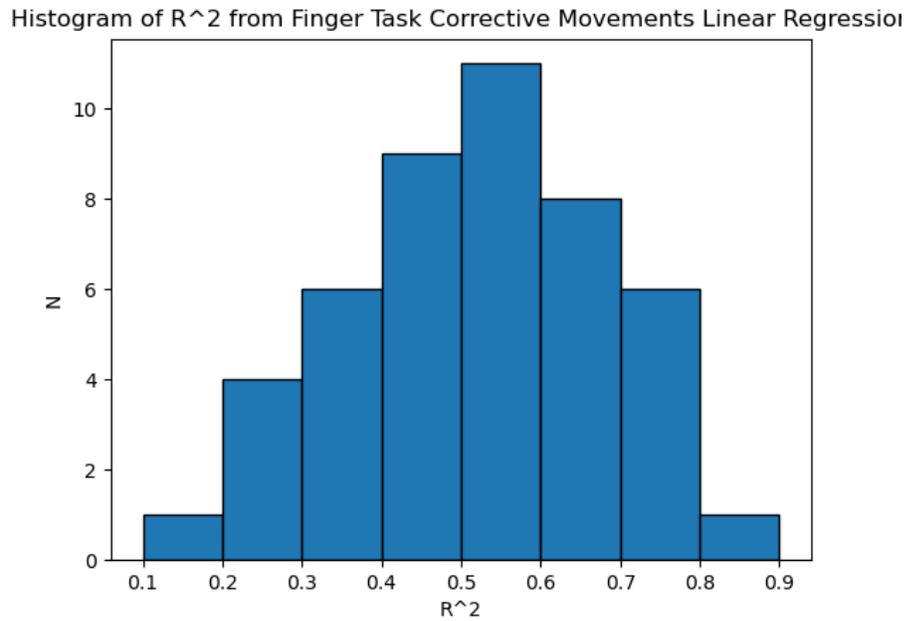


Figure 2.20: Histogram of R^2 values from all number of corrective movements' linear regression which shows the density of the distribution of the R^2 .

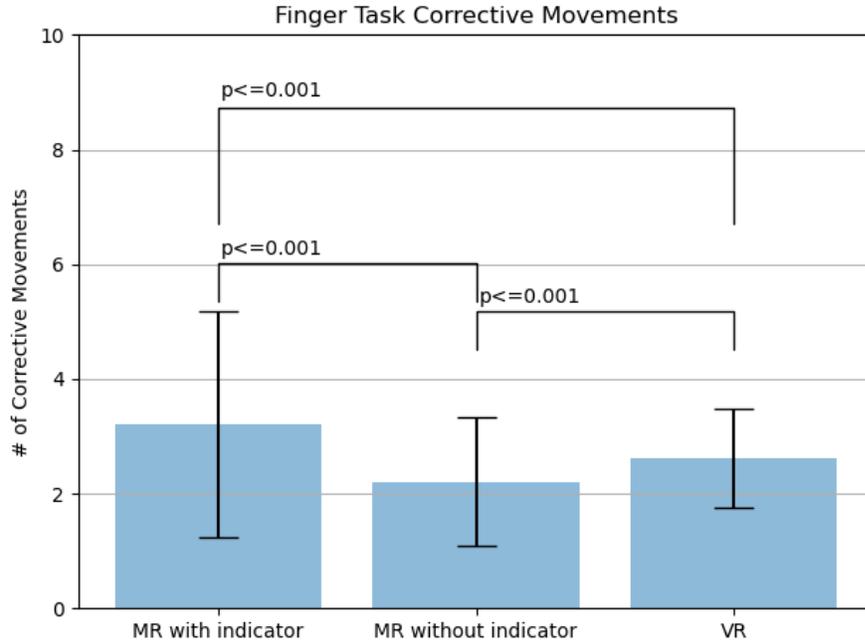


Figure 2.21: Bar plot with error bars for the finger pointing task’s number of corrective movements across all 3 experiment conditions. The mean values for 3 conditions are: “MR with indicator” = 3.21, “MR without indicator” = 2.21, and “VR” = 2.62. lower number of corrective movements mean better performance. “I” shaped error bars stands for the standard deviation, which shows the variability of the number of corrective movements value for each condition. Statistical significance differences between 3 conditions are also shown.

2.5.5 Initial Movement Error

Because there is no clear relation between initial movement error and the corresponding target difficulty. Therefore, initial movement error was computed for every participant on both finger pointing (Fig. 2.27) and cup placement (Fig. 2.28) tasks across all experiment condition, excluding targets’ difficulties. Since the initial movement and the target vectors were normalized before to take their difference, the range of initial movement errors’ results shown below was between 0 and 2. Statistical tests for significant mean differences were calculated with initial movement error as the dependent variable and experiment conditions as the within-subjects factor.

For the finger pointing task, experiment conditions were found to have a significant effect on initial movement error ($p = 0.035$, $F(2, 28) = 3.781$). The “VR” condition

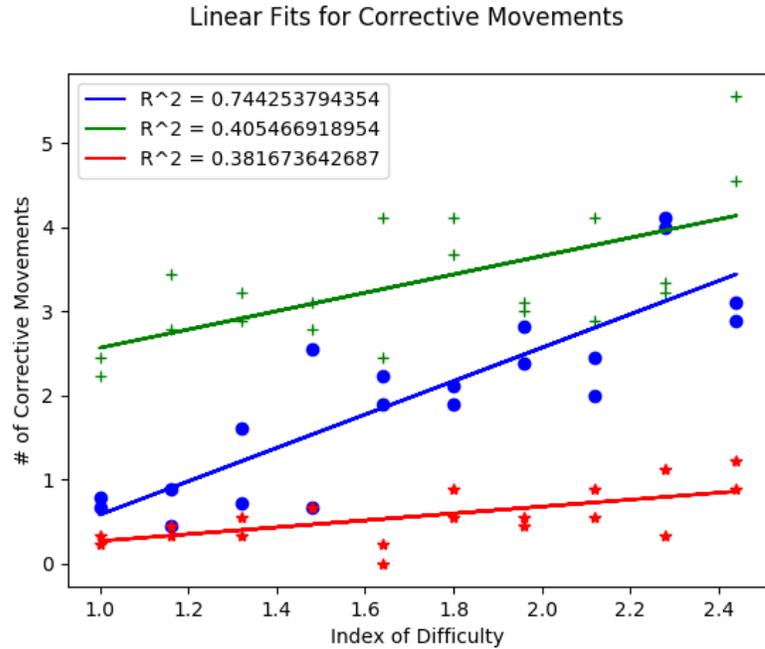


Figure 2.22: Each participant's linear fit from number of corrective movement with corresponding target index of difficulty across all 20 trials.

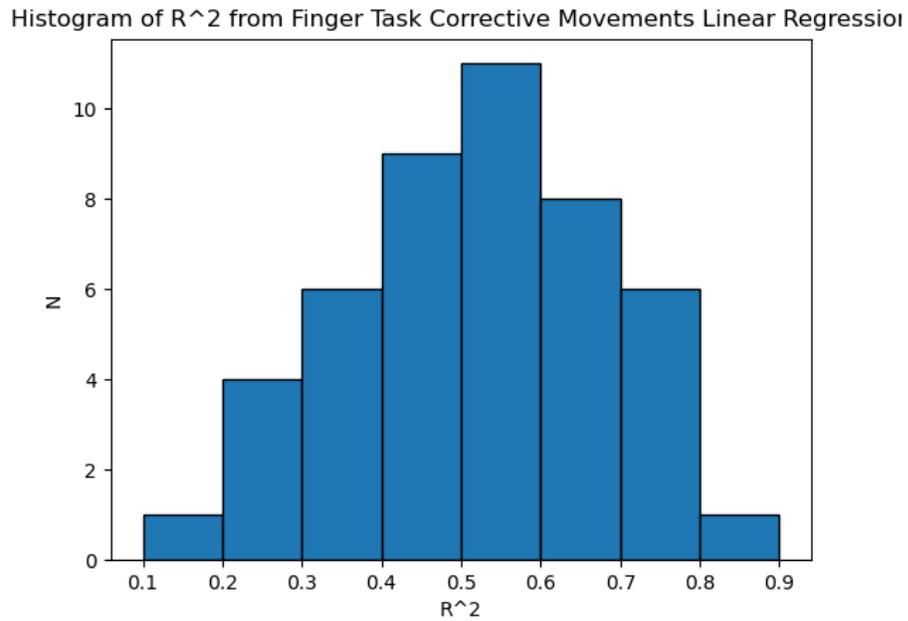


Figure 2.23: Histogram of R^2 values from all number of corrective movements' linear regression which shows the density of the distribution of the R^2 .

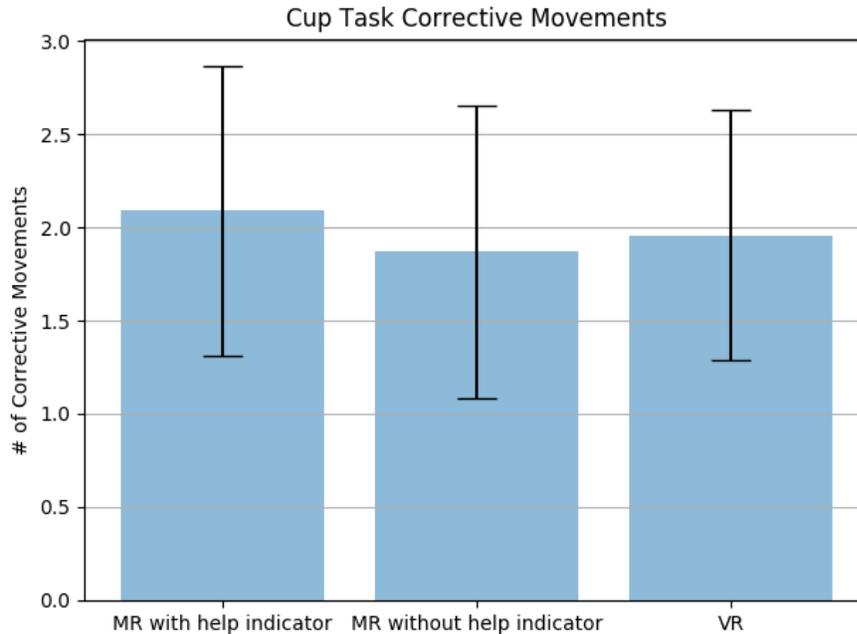


Figure 2.24: Bar plot with error bars for the finger pointing task’s number of corrective movements across all 3 experiment conditions. The mean values for 3 conditions are: “MR with indicator” = 2.09, “MR without indicator” = 1.87, and “VR” = 1.96. lower number of corrective movements mean better performance. “I” shaped error bars stands for the standard deviation, which shows the variability of the number of corrective movements value for each condition

had the significantly lowest mean initial movement error 1.21, and the “MR with indicator” condition’s mean initial movement error was lower than the “MR without indicator” condition.

As for the cup placement task, experiment conditions had no significant effect on initial movement error ($p = 0.96$, $F(2, 28) = 0.039$). In such result, the most angles between participants’ initial movement vector and target vector are around 60 degrees in all experiment conditions (Fig. 2.29).

2.5.6 Peak Velocity

Since target index of difficulty had a significant effect on peak velocity for both finger pointing ($p \leq 0.001$, $F(14, 196) = 115.754$) and cup placement ($p \leq 0.001$, $F(9, 126) = 76.915$) tasks, linear regression were performed. The peak velocity result was also

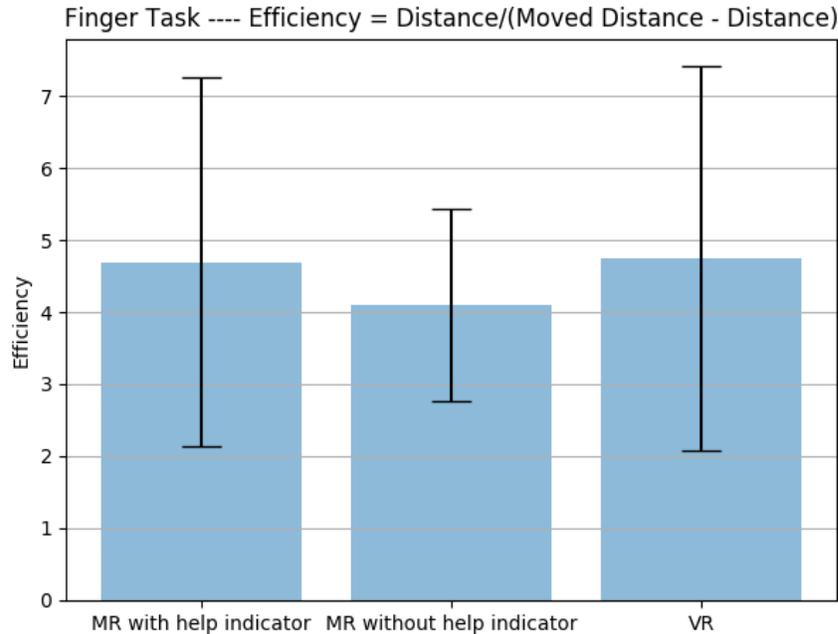


Figure 2.25: Bar plot with error bars for the finger pointing task efficiency across all 3 experiment conditions. The mean efficiency values for 3 conditions are: “MR with indicator” = 4.70, “MR without indicator” = 4.10, and “VR” = 4.75. Higher efficiency means better performance. “I” shaped error bars stands for the standard deviation, which shows the variability of efficiency for each condition.

shown as bar plots with error bars for both tasks. Statistical tests for significant mean differences were calculated using the within-subjects factor (experiment conditions) and dependent variable (peak velocity).

For the finger pointing task, three examples of the linear regression for peak velocity with its target difficulty are shown in Fig. 2.30 and R^2 values of all Linear regression are shown as a histogram in Fig 2.31. Figure 2.32 reports the peak velocity result for every participant in all 3 experiment conditions as a bar plot with error bars. Experiment conditions were found to have a significant effect on peak velocity ($p = 0.0048$, $F(2, 28) = 6.467$). The “MR without indicator” condition’s mean peak velocity (0.39 m/s) was significantly higher than the other conditions. However, the mean peak velocity values of “MR with indicator” (0.34 m/s) and “VR” (0.34 m/s) conditions were not found to be significantly different.

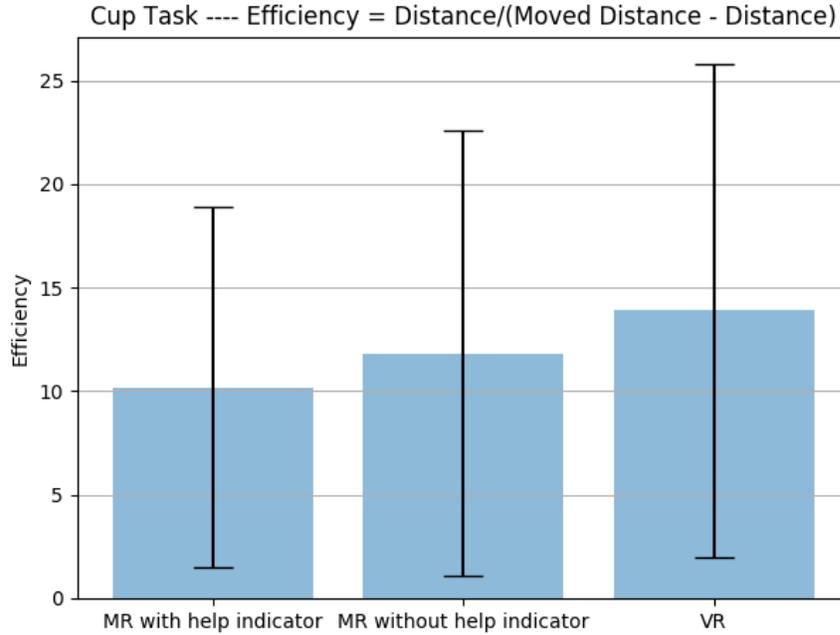


Figure 2.26: Bar plot with error bars for the finger pointing task efficiency across all 3 experiment conditions. The mean efficiency values for 3 conditions are: “MR with indicator” = 10.19, “MR without indicator” = 11.84, and “VR” = 13.91. Higher efficiency means better performance. “I” shaped error bars stands for the standard deviation which shows the variability of efficiency for each condition.

For the cup placement task, the examples from the linear regression are shown in Fig. 2.33 and all linear regression R^2 values are reported as a histogram in Fig. 2.34. Also, a bar plot with error bars of all participants’ peak velocity results is shown in Fig. 2.35. Experiment conditions significantly impacted peak velocity ($p = 0.018$, $F(2, 28) = 4.613$). The highest mean peak velocity is for the “MR without indicator” condition (0.48 m/s). As for “MR with indicator” (0.42 m/s) and “VR” (0.44 m/s) conditions, they were not found to be significantly different.

2.5.7 Questionnaire

We divided all 15 participants into two groups: “Good visual Impression” (the MR display is good) and “Poor visual impression” (the MR display is bad) based on their responses on the questionnaire. The Good visual Impression group included 12 par-

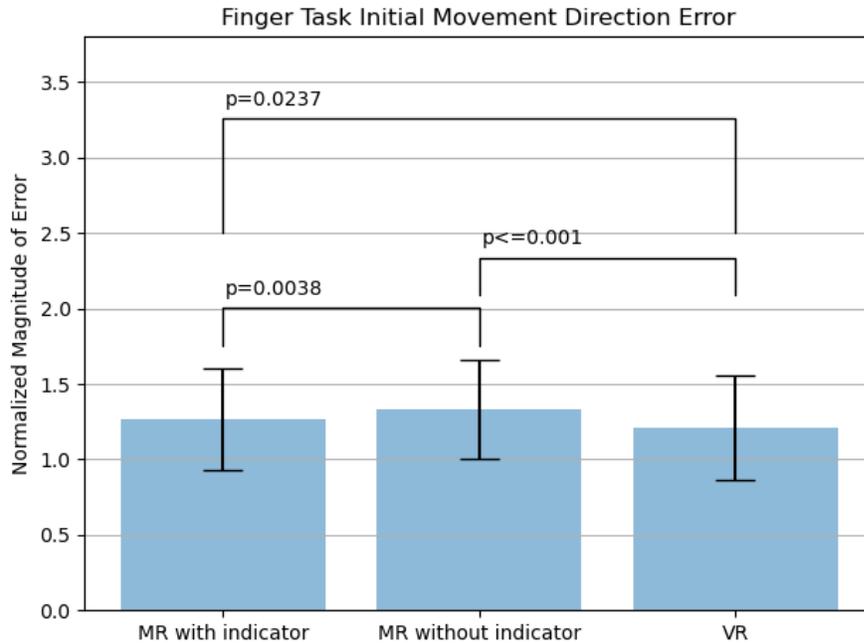


Figure 2.27: Bar plot with error bars for the finger pointing task initial movement errors across all 3 experiment conditions. The mean initial movement error values for 3 conditions are: “MR with indicator” = 1.27, “MR without indicator” = 1.33, and “VR” = 1.21. Lower initial movement error means better performance. “I” shaped error bars stands for the standard deviation which shows the variability of the initial movement errors for each condition. Statistical significance differences between 3 conditions are also shown.

Participants whose mean score is greater than four points, and Poor visual impression groups only have 3 participants whose mean score is less than or equal to four points. Since the responses are highly unbalanced and scant, we performed descriptive statistics analyses: mean value (M) and standard deviation (SD).

For the finger pointing tasks, all six performance value means of the Good visual Impression group were greater than the Poor visual impression group: throughput (Good visual Impression: M = 1.69 bit/s, SD = 0.77 bit/s; Poor visual impression: M = 1.09 bit/s, SD = 0.71 bit/s), end-point error (Good visual Impression: M = 0.008 m, SD = 0.007 m; Poor visual impression: M = 0.014 m, SD = 0.009 m), number of corrective movement (Good visual Impression: M = 2.13, SD = 1.17; Poor visual impression: M = 5.04, SD = 3.22), Efficiency (Good visual Impression: M = 4.85,

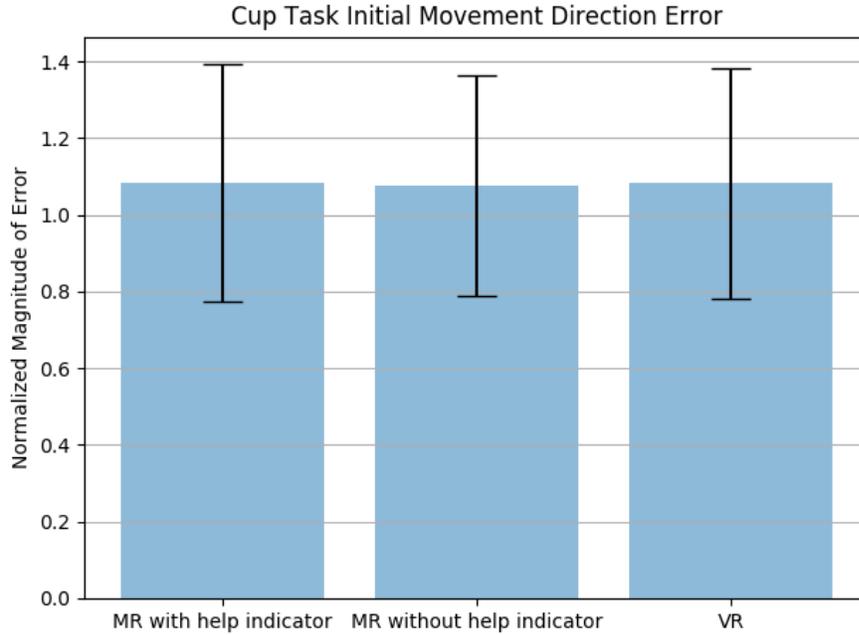


Figure 2.28: Bar plot with error bars for the finger pointing task initial movement errors across all 3 experiment conditions. The mean efficiency values for 3 conditions are: “MR with indicator” = 1.085, “MR without indicator” = 1.077, and “VR” = 1.082. Lower initial movement error means better performance. “I” shaped error bars stands for the standard deviation which shows the variability of initial movement errors for each condition.

SD = 2.58; Poor visual impression: M = 2.59, SD = 1.42), initial movement error (Good visual Impression: M = 1.23, SD = 0.34; Poor visual impression: M = 1.42, SD = 0.29), and peak velocity (Good visual Impression: M = 0.35 m/s, SD = 0.16 m/s; Poor visual impression: M = 0.29 m/s, SD = 0.09 m/s). The bar plots with error bars for these six performance measures are shown in figure 2.36 A.

For the cup placement tasks, the mean values of Good visual Impression group are only greater than Poor visual impression group in three performance measure: throughput (Good visual Impression: M = 1.01 bit/s, SD = 0.41 bit/s; Poor visual impression: M = 0.85 bit/s, SD = 0.37 bit/s), end-point error (Good visual Impression: M = 0.008 m, SD = 0.004 m; Poor visual impression: M = 0.012 m, SD = 0.006 m), and number of corrective movement (Good visual Impression: M = 1.82, SD = 0.94; Poor visual impression: M = 2.60, SD = 1.27). As for efficiency, the

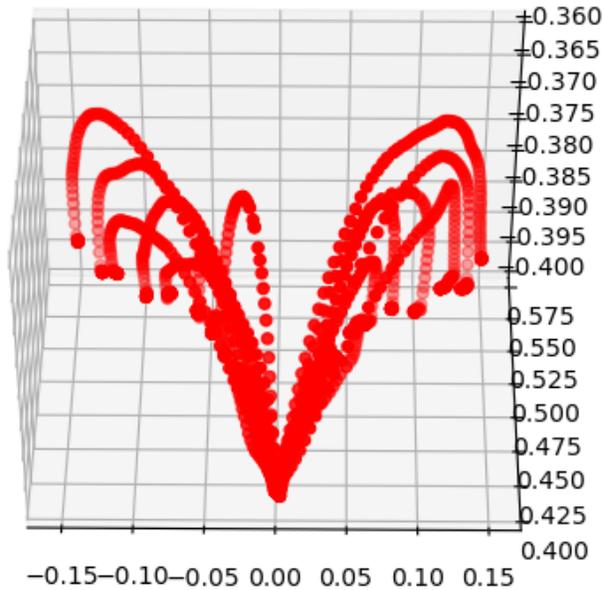


Figure 2.29: An example showed one participant’s movement path for one of the cup placement task.

mean value of Good visual Impression group ($M = 10.32$, $SD = 8.11$) is lower than Poor visual impression group ($M = 13.77$, $SD = 14.78$). Both groups’ mean values for initial movement error (Good visual Impression: $M = 1.07$, $SD = 0.31$; Poor visual impression: $M = 1.05$, $SD = 0.26$) and peak velocity (Good visual Impression: $M = 0.45$ m/s, $SD = 0.19$ m/s; Poor visual impression: $M = 0.46$ m/s, $SD = 0.17$ m/s) are very similar. The bar plots with error bars for these six performance measures are shown below (Fig. 2.36 B).

2.6 Discussion

In general, there appeared to be a trade-off between reaching motion confidence (indicated by throughput, number of corrective movements, and peak velocity) and

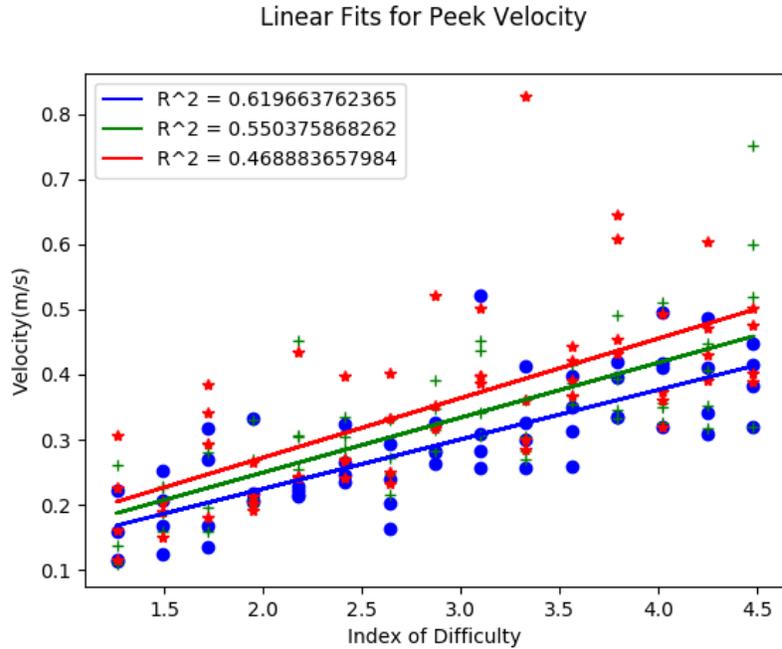


Figure 2.30: Each participant’s linear fit from peak velocity with corresponding target index of difficulty across all 60 trials.

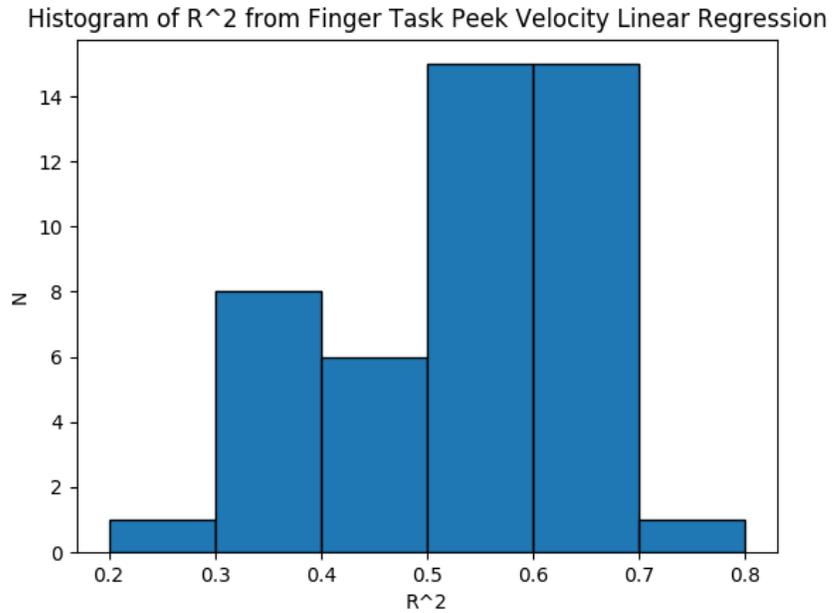


Figure 2.31: Histogram of R^2 values from all peak velocities’ linear regression which shows the density of the distribution of the R^2 .

accuracy (indicated by end-point error and initial movement error). Furthermore, compared to the finger pointing task, the cup placement task’ experiment conditions

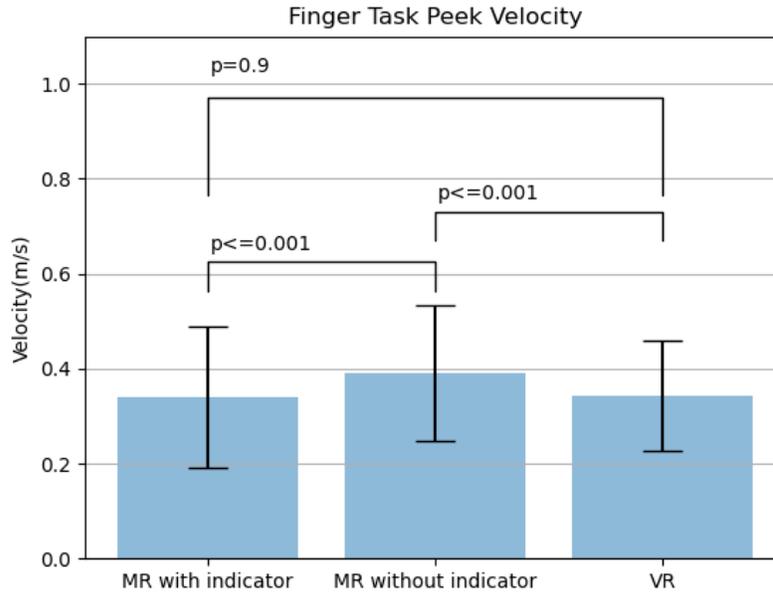


Figure 2.32: Bar plot with error bars for the finger pointing task’s peak velocity across all 3 experiment conditions. The mean values for 3 conditions are: “MR with indicator” = 0.34 m/s, “MR without indicator” = 0.39 m/s, and “VR” = 0.34 m/s. Higher peak velocity mean better performance. “I” shaped error bars stands for the standard deviation which shows the variability of the peak velocity value for each condition. Statistical significance differences between 3 conditions are also shown.

tended to have a significant effect on fewer measures of reaching performance.

2.6.1 Finger Pointing Task

The results of the finger pointing task show participants under the MR environment without the finger indicator exhibited faster and smoother trajectories than under the MR environment with indicator and VR environment. The mean throughput for the “MR without indicator” condition was 1.23x higher than the “MR without indicator” condition and 1.22x higher than the “VR” condition. Also, the number of corrective movements for the “MR without indicator” condition was about 0.69x lower than the “MR with indicator” condition and 0.84x lower than the “VR” condition. Similar to the throughput result, peak velocity for the “MR without indicator” was

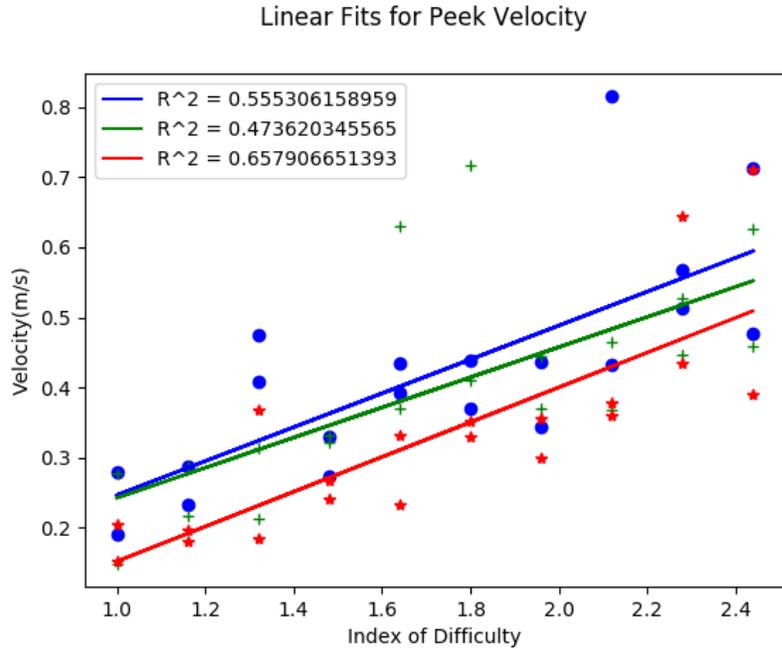


Figure 2.33: Each participant’s linear fit from peak velocity with corresponding target index of difficulty across all 20 trials.

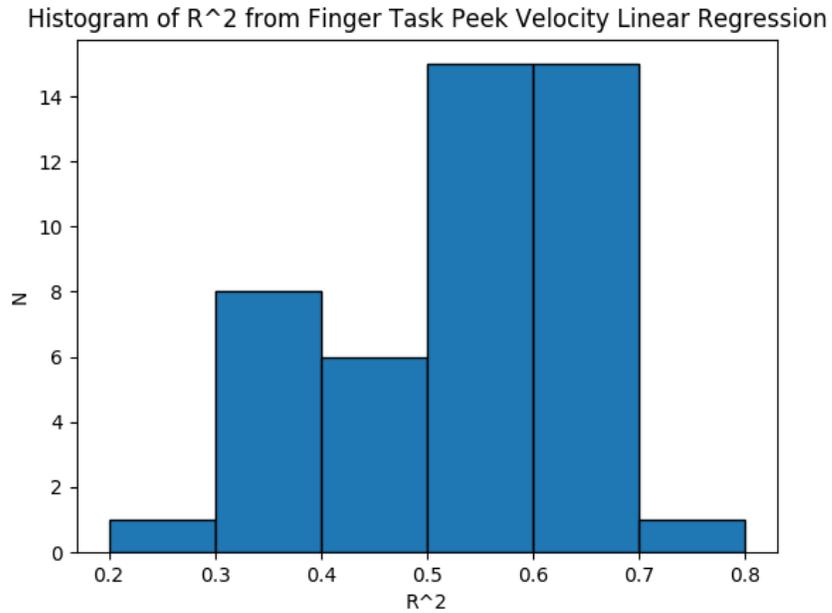


Figure 2.34: Histogram of R^2 values from all peak velocities’ linear regression which shows the density of the distribution of the R^2 .

approximately 1.15x higher than the “MR with indicator” and “VR” conditions.

Although the mean efficiency for the “MR without indicator” condition was a bit

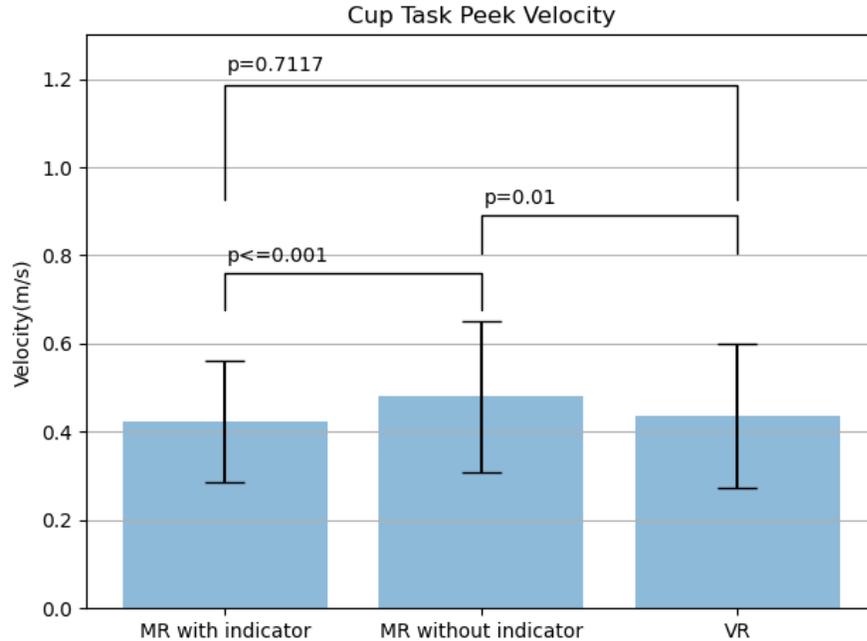


Figure 2.35: Bar plot with error bars for the finger pointing task’s peak velocity across all 3 experiment conditions. The mean values for 3 conditions are: “MR with indicator” = 0.42 m/s, “MR without indicator” = 0.48 m/s, and “VR” = 0.44 m/s. Higher peak velocity mean better performance. “I” shaped error bars stands for the standard deviation which shows the variability of the peak velocity value for each condition. Statistical significance differences between 3 conditions are also shown.

lower than the other two conditions, this trend did not reach statistical significance.

On the other hand, the results of end-point error and initial movement error showed that participants under the “MR without indicator” condition had a greater difficulty locating the target points than the other two conditions. The mean end-point error for the “MR without indicator” condition was 2.60x higher than the “MR with indicator” condition and 3.25x higher than the “VR” condition. As for initial movement error, the mean value for the “MR without indicator” condition was 1.05x higher than the “MR with indicator” condition and 1.10x higher than the “VR” condition.

We observed that the mean throughput and peak velocity values between MR with end-point indicator and VR conditions were almost the same. However, the mean end-point error for the “VR” condition was 0.80x lower than the “MR with

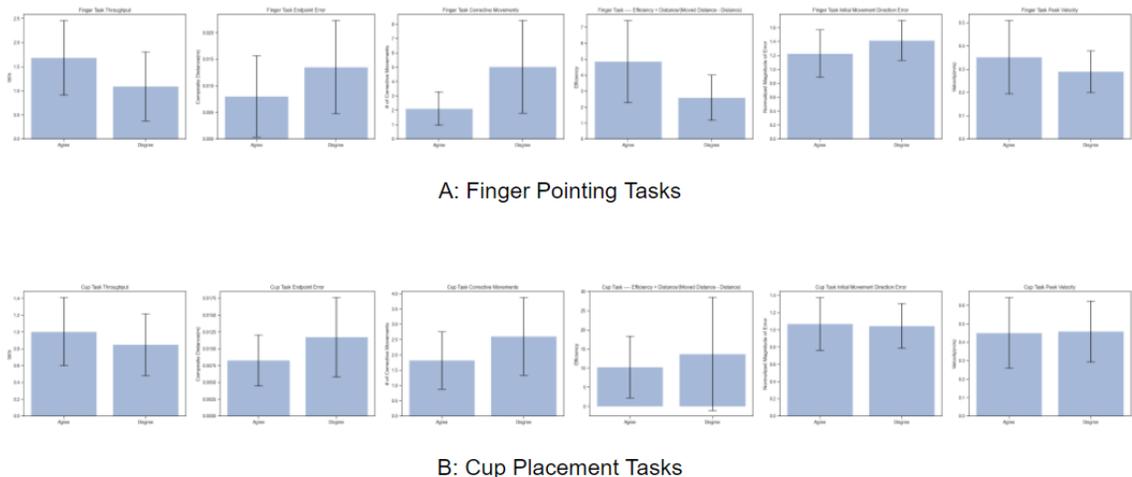


Figure 2.36: The 12 figures are bar plots with error bars of six performance measures for finger pointing tasks (top) and cup placement tasks (bottom). The six performance measures from left to right are: throughput, end-point error, number of corrective movements, efficiency, initial movement error, and peak velocity.

indicator” condition. Also, the initial movement error for the “VR” condition was 0.95x lower than the “MR without indicator” condition. From this comparison, we can see participants’ task performance under these two conditions were very similar where the “VR” condition was slightly better than the “MR with indicator” condition.

By comparing with related Fitts’ task literature, we found the performance of our “MR with indicator” condition had a 1.29x increase in throughput and 2.01x decrease in end-point error compared with [29]’s best virtual hand techniques. The discrepancy may be due to the fact that this prior work used a non-colocated computer monitor as a display, where as our experiment used an immersive MR display (co-located). There is evidence that human point-to-point reaching performance under the co-located experiment condition is better than under non-colocated [5].

By comparing with [5]’s VR experiment, our “VR” condition’s results had a 2.38x decrease in throughput and a 1.81x decrease in end-point error. This further supports the idea that there could be a trade-off between throughput and end-point error in

this comparison.

From the result of the questionnaire, we can see a trend from the mean values where participants with positive impressions of MR perform better than those with negative impressions. However, we have not shown the difference between these two groups is statistically significant or not due to the unbalanced and limited size of sample. Future work needs to be done to clarify that.

2.6.2 Cup Placement Task

Many results for the cup placement task did not reach the statistical significance, but some interesting trends were observed.

Similar to the finger pointing task, participants under the “MR without indicator” condition did have faster and smoother reaching motion than “MR with indicator” and “VR” conditions for the cup placement task. The mean peak velocity for “MR without indicator” condition was 1.14x higher than “MR with indicator” condition and 1.09x higher than “VR” condition. Although the cup placement task’ throughput and the number of corrective movement measurements did not show that experiment conditions had a significant effect on these measures, we observed that they also have the same trend on the results as the finger pointing task (Figs. 2.13, 2.16, 2.21, and 2.24).

We also observed that participants had the most difficulty reaching the target location in an MR environment when not presented with an end-point indicator. From end-point error results, we found that the mean value for “MR without indicator” condition was 1.25x higher than “MR with indicator” condition and 1.43x higher than “VR” condition.

Another observed trend was that participants had marginally better performance under the “VR” condition than the “MR with indicator” condition. The end-point error for the “VR” condition was 0.88x lower than the “MR with indicator” condition.

Also, the mean peak velocity for the “VR” condition was 1.05x higher than the “MR with indicator” condition.

For initial movement error, we observed that the mean initial movement errors for all three conditions are very similar and closed to 1 (“MR with indicator”: 1.085, “MR without indicator”: 1.077, and “VR”: 1.082), which indicated the angle between the initial movement and target vectors was off by approximately 60 degrees. We discovered that 13 out of 15 participants had this type of initial trajectories that moved toward the target after picking the cup up. Additionally, 8 out of 15 participants’ most cup placement movements are like this (Fig. 2.29).

For the cup placement task, our “MR with indicator” condition had a 1.39x increase on throughput and 1.54x decrease on end-point error when compared with [29], in which participants performed a similar cup placement task as the one in the current work. The possible reason for this increased performance is the same as the reason mentioned in 2.6.1, which may be the difference between co-located and non-located display.

From the result of the questionnaire, we can see a trend from the mean values on three performance measures (throughput, end-point error, and number of corrective movements) where participants with positive impressions of MR have better performance than those with negative impressions of MR. However, the other three performance measures (efficiency, initial movement error, and peak velocity) do not show that trend. Additionally, we have not shown that the difference between these two groups is statistically significant due to the unbalanced and limited sample size, which needs to be addressed in future work.

2.6.3 Trade-off between Reaching Motion Confidence and Accuracy

A key finding was evidence for a trade-off between performance measures that indicate reaching motion confidence (throughput, peak velocity, and number of corrective movements) with the performance measure that indicates accuracy (endpoint error). Specifically, movement confidence measures were highest in the MR without indicator condition for the finger pointing task, while accuracy was lowest. Although this did not support our hypothesis that displaying the indicator in the MR condition would result in highest reaching performance on all measures, this may indicate that an unintended consequence to displaying the indicator (it was done to replace missing depth perception cues which showed in Figure 2.6). It is possible that because participants had impaired depth perception in the MR without indicator condition, they were unable to follow our instruction to move both quickly and accurately to the target and prioritized moving fast at the cost of accuracy. Therefore, participants under the “MR without indicator” condition had the faster and smoother reaching motions while the accuracy was reduced significantly.

2.6.4 MR and VR Application Design Implications

Based on the current result, there are some MR/VR application design recommendations can be considered. When both MR and VR are under consideration, MR is recommended if precise end-point accuracy is not required. If the features of MR are necessary, then display of end-point indicators (indicator can be various based on the application) is recommended. On the other hand, if MR is not necessary and end-point error is critical, then a VR-based environment design may be most suitable.

2.7 Future Work

In this study, we evaluated the 3D Fitts' point-to-point reaching performance in 3 MR and VR conditions. However, we had a limited sample size of 15 participants due to the pandemic, which leads to reduced statistical power and less conclusive results. Therefore, more participants are needed in the future. Also, it is still unclear why most performance measures of the cup placement task had no statistical significance. Such question might be answered by investigating it in the future. Additionally, the question “how the participants' visual impression about certain display devices affects the point-to-point reaching performance” can be investigated in the future. At last, the current study was designed only to give participants limited time to practice. Results may differ if participants have more time to practice and be familiar with the HoloLens device.

Chapter 3

Human-Swarm Teaming Through Multi-user Interface With Mixed Reality

Human-swarm teaming (HST) refers to the collaboration of multiple humans with a large group of robotic agents, which is a challenging and novel application of immersive MR displays and virtual environment design. As part of the DARPA Offensive Swarm-Enabled Tactics (OFFSET) project [35], We developed an innovative multi-modal MR interface for simulated aerial drone HST that is integrated with a consensus decision making algorithm to assign tasks to the drones [36, 37], and a sensorimotor haptic interface for users to input commands and receive feedback through implanted and surface EMG signal [38, 39, 40].

Our interface was designed to facilitate the collaboration of three tactical human tactician roles with 150 simulated semi-autonomous aerial drones on the Case Western Reserve University campus (Fig. 3.1). We also demonstrated this MR sensorimotor HST interface with a veridical robotic swarm at DARPA OFFSET Field Exercise 3.

The scope of this chapter is on the design of the MR user interface and its integra-

tion with the other components of our innovative HST system, including the consensus decision algorithm and sensorimotor haptic interface, which were developed by our collaborators on this project.

This chapter is in preparation for publication to the journal *Virtual Reality* [41].

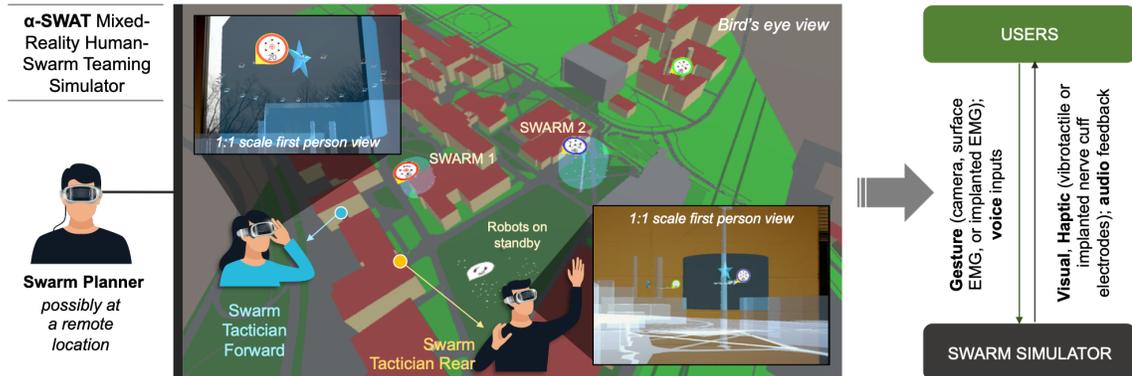


Figure 3.1: Conceptual illustration of the multi-user visuo-sensorimotor interface for human-swarm interface. Users–Swarm Planner (SP), Swarm Tactician-Rear (STR), and Swarm Tactician-Forward (STF)–communicate and collaborate with the swarm at the levels of individual robot, subset, and the entire swarm via vision- and sensorimotor-based interaction. DARPA Distribution Statement "A" (Approved for Public Release, Distribution Unlimited).

3.1 Background

This section will focus on topics that relevant to human-swarm teaming (HST), human-swarm interactions (HSI), and hand-based gesture/MR used in our human-swarm teaming system.

3.1.1 Human-Swarm Teaming and Human-Swarm Interactions

Human-swarm teaming refers to the simultaneous interaction between multiple humans and swarms of autonomous robots. Its precursor, human-swarm interaction

(HSI) refers to the interaction between a single human and a swarm of autonomous robots on a task.

In general, there are five roles for humans in human-robot-interactions [42], which include: supervisor, operator, teammate, bystander, and mechanic. Supervisor can oversee and evaluate the overall situation and give the high level goals of the missions. As for the operator, such a role is required to control and monitor low level tasks by directly commanding robots. For the teammate role, they command robots with high level tasks without changing the supervisor's overall goal. In our application, DARPA defined three personas (Swarm Planner, Swarm Tactician Rear, and Swarm Tactician-Forward) that are the combination of the supervisor, operator, and teammate roles. Swarm Planner (SP) plans missions at a potentially remote location prior to the swarm deployment. Swarm Tactician Rear (STR) oversees a mission in real-time from a location adjacent to the operation area. Swarm Tactician Forward (STF) performs tasks with the robots within the mission operation area.

In HST/HSI, both human operators and the swarm work together to achieve success in missions[43]. An ideal HST interface is designed to optimize human factors so that human operators can handle dynamic changes and efficiently manage the swarm's ability to carry out predefined tasks (referred to as swarm tactics) [43] [44]. Adams categorized several human factors that are important for human-robotic interfaces design and developing [45]. These human factors include human decision making, situation awareness (SA), vigilance, workload levels, and human error. Human decision making processes typically happened rapidly under different environments and conditions. Situational awareness refers to the operator's ability to appropriately understand the status of robots and the environment. Vigilance refers to the ability to sustain attention and keep up with the operation over time. Workload can be either mental or physical workload, which can affect the operator's ability to complete tasks properly. As for human error, it can cause task failures and reduce efficiency and

safety. All these human factors are related to each other. For example, Appropriated human decision making can reduce human error. Situational awareness and apposite workload level can help to maintain the operator’s vigilance level. A reduced SA may lead to a low level of vigilance and also adversely affect the decision making process. Additionally, too much information and/or too many tasks (high workload) can negatively affect vigilance level, SA, and decision making during the operation. Consistent with the literature, we developed an HST interface to facilitate efficient decision making with our consensus decision making algorithm, maximize situational awareness with our immersive MR interface approach, and maintaining vigilance level with our feedback system.

User interfaces are necessary to achieve HST with larger amounts of robots. Several UI modalities have been developed for HSI/HST, such as a point-and-click graphical user interface (GUI) developed for teleoperation and remote monitoring of a robotic swarm for detecting radiation sources [46]. Others developed an HSI interface with a virtual avatar that verbally reports information gathered by swarm [47]. Another HST interface using haptic device and gesture detection was developed for navigation purpose to assist fire-fighters in a burning building [48].

3.1.2 Human Inputs for Human-Swarm Teaming

Hand gestures are an intuitive way to represent certain important messages. Many researches showed the use of hand gestures as an effective way for communication between humans and robots in HST/HSI [49, 50, 51]. Specifically, hand gesture inputs are more user friendly and effective compared to the conventional methods like keyboard [50]. One study demonstrated the success of commanding aerial swarms using hand-based gestures through a VR system [9]. Similarly, another study showed using hand gestures detected by Leap Motion 3D cameras and voice commands to define drone flight paths and plan swarm missions [52, 53].

Even with the benefit of hand gestures in HST/HSI, it may not be possible for users to perform gesture inputs when they need to use their hands to carry objects or work on multiple tasks during HST/HSI. Additionally, users using hand gestures input may also need haptic feedback which requires additional physical devices, which might not be ideal for outdoor use. Studies have been developed the portable glove-based systems that control swarm with inertial sensors and provide haptic feedback with vibrotactile motors [54, 55]. Additionally, EMG signal can also be used as human input. Skin-surface EMG was used for gesture-based drone control [56], which allows users to give gesture command by only contracting muscles. Unlike Skin-surface electrodes, implanted nerve interfaces can acquire high quality EMG, provide direct tactile feedback through electrical stimulation, and do not require donning and doffing. Our team’s prior work shown that a participant with upper limb amputation with percutaneously implanted direct nerve interface could control a prosthetic hand with EMG signal [57]. Our project will utilize camera-based, skin-surface EMG based, and implant EMG based hand gesture controls for different roles in HST.

3.1.3 Mixed Reality’s capabilities for Human-Swarm Team- ing

In recent years, VR was introduced for many HST/HSI studies [58, 8, 9]. However, VR head-mounted displays obstruct the user’s view of the physical environment, which is not suitable for users who need to see their surroundings or interact with other people. On the contrary, the capabilities of MR allow it to present swarms location and activities to the user in the operation area, even if swarms are behind buildings or walls [59]. With the advantages of MR, users will have the ability to receive a wealth of information from both virtual and physical environments. A study has shown that immersive displays can improve users’ spatial understanding compared to standard monitors where spatial understanding is an essential part of users’ SA [60]. Another

study demonstrated the use of MR displays to guide a swarm of 20 robots to a target in a physical environment [61]. Additionally, MR interfaces have also been developed to allow users to visualize swarm status and assign tasks to individual robots [62]. However, no MR interfaces have been developed that can adapt to multiple human roles. Therefore, with the advantages of MR, our immersive MR interface is ideal for HST in both indoor and outdoor environments.

3.2 Methods

The following section described the hardware used in the project, software design, participants, and experiment procedure.

3.2.1 Hardware

Up to three HoloLens devices could be used simultaneously in the developed interface; one for each of three personas (SP, STR, STF). A Windows 10 Pro Alienware 15 R3 laptop was used to run the central server which connected with all the HoloLens and other devices like: a PC ran Robot Operating System (ROS), and another PC ran Matlab scripts for data acquisition, and a Ripple Grapevine device that read raw EMG data (Fig. 3.2).

3.2.2 Software Design

The software was designed and developed by team members of this project. The following section will describe the MR interface (we called it Clients) and server like Unity simulation (we called it Server) development I worked on.

To easily establish communication between Server (Unity simulation of the swarms' action) and Clients (HoloLens), the Unity3D game engine was used to develop the user interface for different personas and Server. C# was the programming language

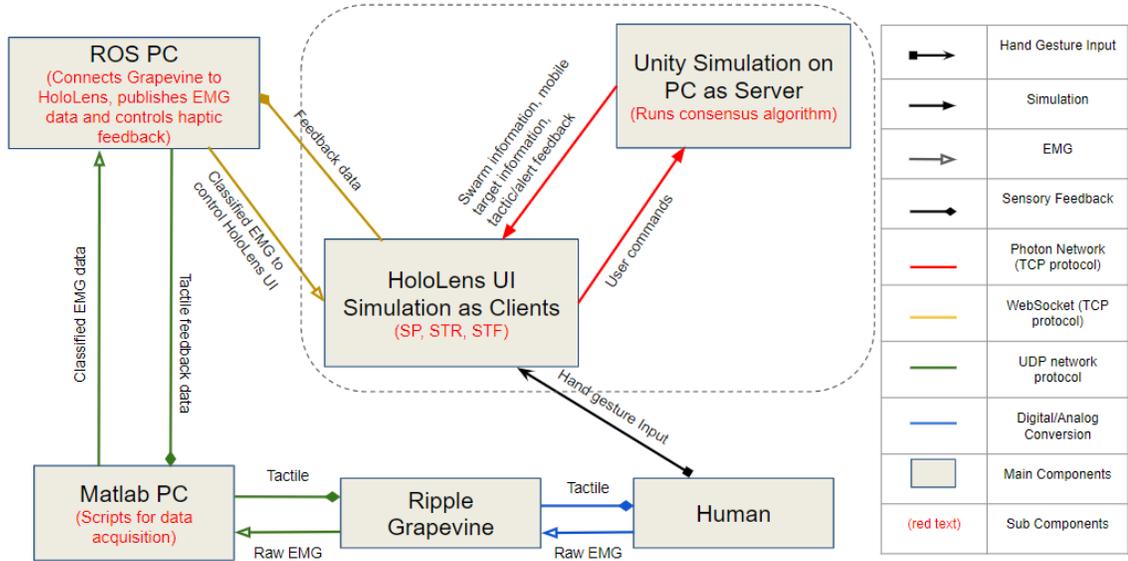
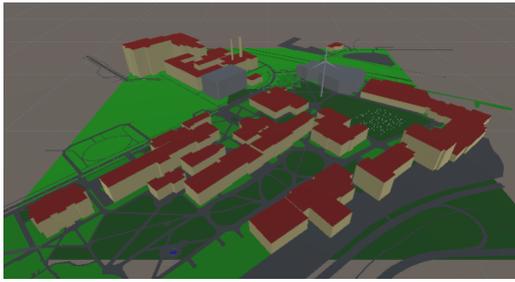


Figure 3.2: A system block diagram of components, communication contents and protocols. Dotted frame included the part that this thesis focused on. DARPA Distribution Statement "A" (Approved for Public Release, Distribution Unlimited).

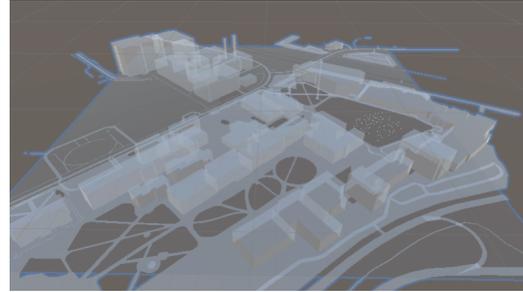
used in the implementation.

A HoloLens mixed reality display was used to visualize the environment and swarm at two different scales and perspectives: one calls miniaturized map, and another is 1:1 map (Fig. 3.3). Both 3D maps of the part of Case Western Reserve University campus were sourced from OpenStreetMap.org, and were also scaled and edited using JavaOpenStreetMap Editor, OSM2World, and Blender software applications which Unity game engine can read. Although the map data did not include building heights, we referenced height measurements from campus blueprints and adjusted them to visually match the physical buildings. One difference between these two maps is that the 1:1 map had the same size as the physical environment while the miniaturized map was shrunk down to a smaller size of 2 by 2 meters. Therefore the miniaturized map can present the overall situation of the whole area. Another difference is that the miniaturized map was colored while the first person map was made to be semi-transparent (3D texture alpha channel = 0.33). This was done because the semi-transparent 1:1 map (aligned with physical environment) enables user's ability to

have x-ray vision and helps users to locate the swarm beyond their line of sight.



(a) Miniaturized map



(b) 1:1 map

Figure 3.3: 3D CWRU campus model in two different type. DARPA Distribution Statement "A" (Approved for Public Release, Distribution Unlimited).

Map Calibration

For all three personas, virtual 3D maps were calibrated to align with the physical environment. To do that, world anchors were used. It allows users to define, save, and load the world anchors' location, so the virtual objects can stay at the defined veridical location across sessions. Each persona needs to start the Client at a predefined location and orientation that exists in both virtual and physical environments. Then we place two world anchors with the same appropriate distance in front of and behind the initial location.

User Inputs and Feedback

Users can interact with virtual objects by performing a selection input. There are three different methods to trigger a selection input, but all of them require the user to point a "gaze cursor" at the preferred virtual objects and then trigger the selection input in one of 3 methods. The "gaze cursor" is a small dot that always stays at the center of users' field of view and keeps on top of any virtual or physical surfaces. "Gaze cursor" can programmatically scale and deform when it interacts with either the miniaturized map or menus.

The first selection input is using HoloLens' "Air Tap" hand gesture recognition, it requires users to put their hand in front of the device's front cameras and tap the index finger and thumb together to trigger the selection functions. Another selection input needs users to wear multiple skin-surface electrodes on their performed arm and hold a fist hand gesture for the system to read the processed EMG signal from corresponding contraction muscles (surface-EMG). The last selection input is very similar to the second one, except it can be performed by people with below-elbow amputation, and the EMG signal reading is from the implanted nerve cuff electrodes in the user's forearm (implanted-EMG). For both EMG signal readings, when the processed data reached the pre-defined threshold (0.2) and time (0.5 seconds) on the Client side, the system will trigger the selection functions (Fig. 3.4).

The system provides different types of feedback. Audio feedback can be triggered by selection functions to help users confirm that the selection inputs were properly registered. Two different sound effects can be triggered by either standard selection inputs or when users received a notification alert message. Another feedback for able-bodied users can be delivered by vibrations on users' fingers with a vibrotactile motor glove worn by users. The third type of feedback is delivered through the sensory stimulation of the implanted nerve cuff electrodes for amputee users. These last two types of feedback and the audio feedback are triggered with the same function, which reinforces the intensity of the feedback.

Tactics

The experiment scenario includes four tactics. Three of them can be planned by SP/STR, which are "Aggregation", "Dispersion", and "Secure Boundary". All three tactics require the swarm to gather up and form a formation at a pre-defined location/area. The only difference is "Aggregation" needs the swarm to gather at one point, "Dispersion" asks the swarm to disperse in the target area evenly, and "Secure



Figure 3.4: Left: Amputee participant was using his implanted nerve interface to configure swarm tactics. Right: Able-bodied user was using his skin surface EMG to configure swarm tactics. DARPA Distribution Statement "A" (Approved for Public Release, Distribution Unlimited).

Boundary" requires the swarms to form as a ring at the boundary of the pre-defined target area. There is another tactic called "Surveillance". It triggered by the swarm (let us call it "swarm a") that captured a simulated "Moving Target" on the Server side, then a new swarm of 5 drones from "swarm a" will start following the "Moving Target" to collect information and alert only STR persona. After STR received this alert message, human input is required to either ignore or disable the "Moving Target".

UI Design for Three Personas

There are three personas for Clients: SP, STR, and STF. Each persona has different user interfaces. After users started the application on HoloLens, they got to choose their roles (Fig. 3.5). As for Server, an instruction of pressing the "S" key for launching the Server application. The Server must run the simulation first before Clients join.

Swarm Planner Swarm Planner (SP) persona acts as a supervisor who pre-plans the high-level tactics and oversees the swarm actions. We assumed that SP plans a

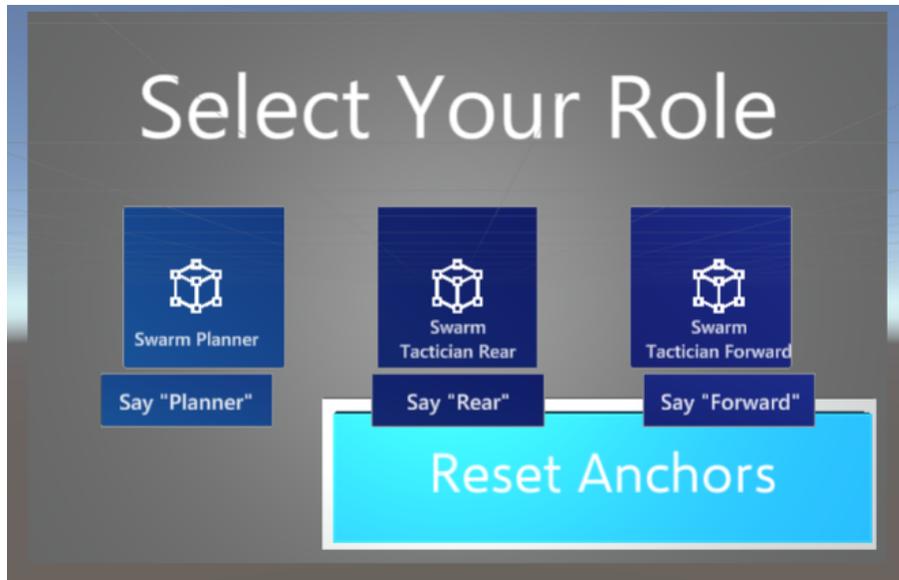


Figure 3.5: User select menu. DARPA Distribution Statement "A" (Approved for Public Release, Distribution Unlimited).

mission from a remote location before the swarm is deployed. For this role, the user is presented with a 3D map of the operation area that is scaled down and positioned near waist-level to appear like a table-top model (Fig. 3.6).

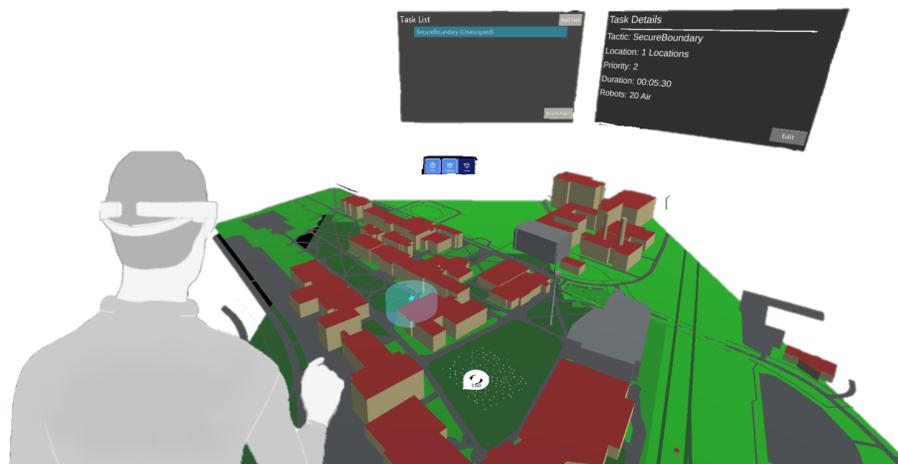
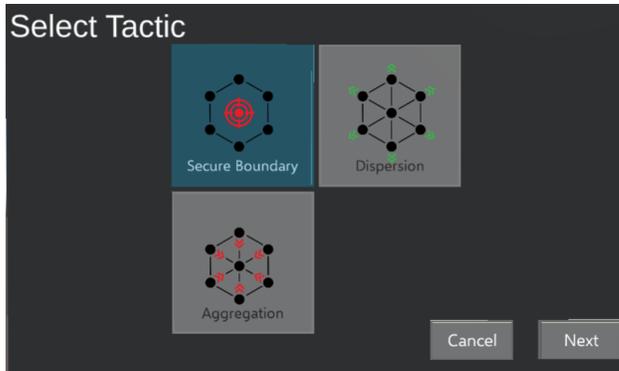


Figure 3.6: An example of tactic planning process with our MR user interface and miniaturized map. The mid-top floating window shows a list of configured tactics, and the right-top window is where the tactics' configuration and detail display happened. DARPA Distribution Statement "A" (Approved for Public Release, Distribution Unlimited).

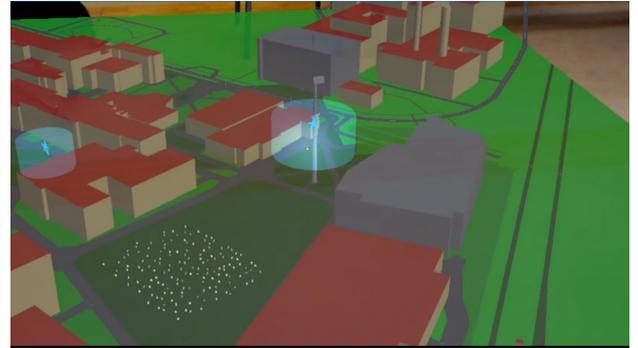
The tactics planning process for SP took place after opening the tactic list window was positioned on the opposite side of users and above the miniaturized map. By selecting the “Add Task” button, users can select a tactic to configure from an array of possible choices on the pop-up window on the right side (Fig. 3.7 a). Then users mark the target location of the chosen tactic on the miniaturized map as a blue rotating star by placing the “gaze cursor” on the preferred location of the map and performing selection input. For “Dispersion”, and “Secure Boundary” tactics, users need to define the area of interest (centering the star) by scaling a transparent blue cylinder by moving the “gaze cursor” and another selection input (Fig. 3.7 b). The next step is to define the duration of a swarm on such tactic in the next menu (Fig. 3.7 c). Users can configure the duration by selecting either “-5”, “-1”, “+1” and “+5” of the hours, minutes, and seconds options. Finally, users can define the number of drones on a swarm for a tactic that has a similar interface as the duration configuration menu (Fig. 3.7 d).

After tactics are planned, SP can go back to view and edit each tactic in detail. After all tactics are defined, the tactic list is transmitted to the Swarm Tactician-Rear’s user interface by triggering the button labeled “Send to STR” at the bottom right corner of the task list (Fig. 3.6). Haptic feedback is provided to the user after the list is successfully transmitted.

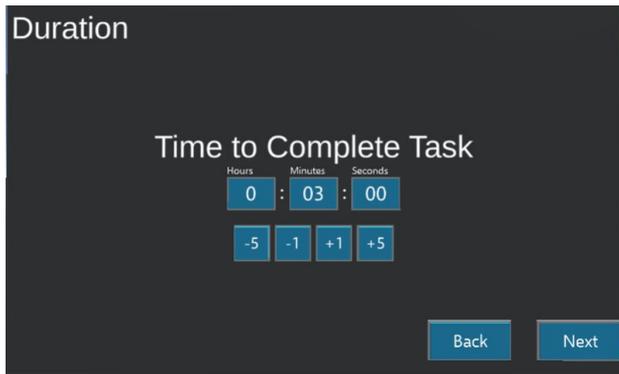
Swarm Tactician-Forward STF persona had a completely different user interface compared with SP’s. By the nature of STF, users need to be actively involved in the missions at the actual operation area with swarms. To suit this role, the user is presented with a semi-transparent version of the 3D map that is overlaid upon the real environment and viewed through the first-person perspective. The rationale for this design is to augment the user’s situational awareness of the swarm and real environment beyond their line of sight. Analogous to x-ray vision, this interface



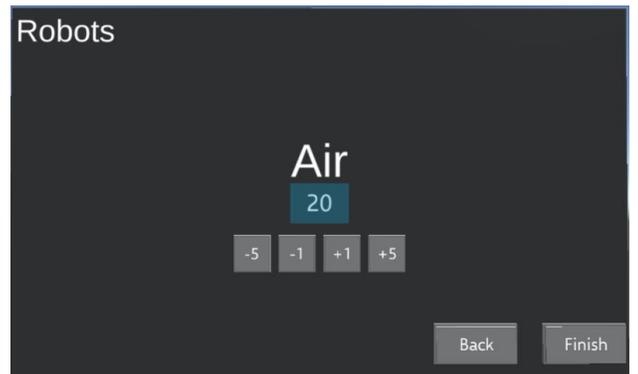
(a) Tactic configuration menu.



(b) Tactic location (cyan star) and area (cyan cylinder) definition.



(c) Tactic duration configuration menu.



(d) Drone number configuration menu.

Figure 3.7: Tactic configuration process for SP and STR role. DARPA Distribution Statement "A" (Approved for Public Release, Distribution Unlimited).

enables STF to maintain visualization of tactic status even when swarms are located inside or between buildings. This concept is demonstrated in figure 3.8, which shows the mixed reality 3D map closely co-located with the real environment, which is visible through a window in the room. Another Figure 3.9 shows that the map overlay can visualize buildings that would otherwise be obstructed by a wall to the right of the window. However, since HoloLens only captures the right eye's holographic output, all the video capture's 1:1 map is a bit off from the physical environment. Additionally, swarm tactic status is visualized as the circular icon in Figure 3.8, which indicates that a swarm of 20 drones reached consensus and are actively implementing the Secure Boundary tactic at a rooftop directly in front of the user. It also shows each of the 20 simulated aerial drones forming a circle around the semi-transparent cyan cylinder

that defines the tactic target area.

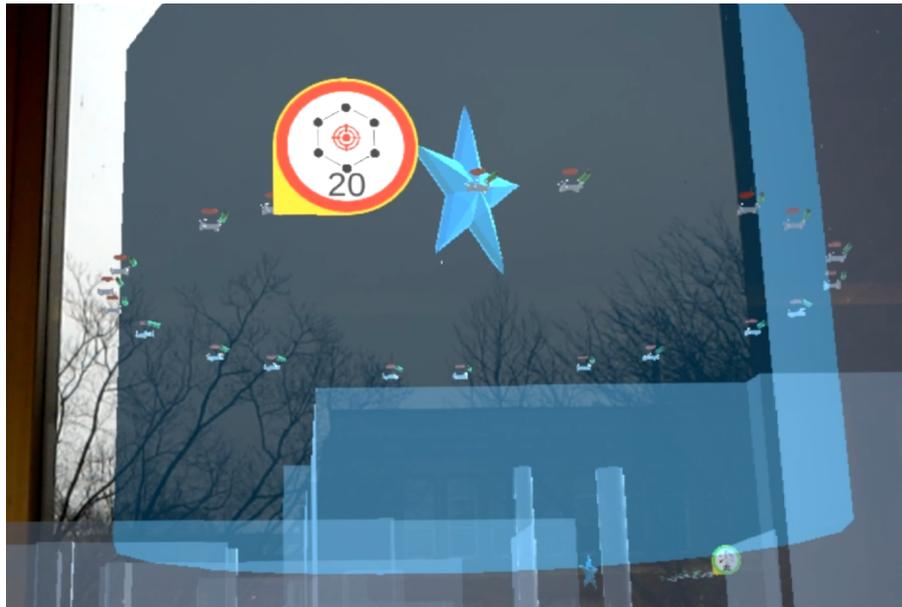


Figure 3.8: First person view of swarm actions and MR environment for STF. DARPA Distribution Statement "A" (Approved for Public Release, Distribution Unlimited).



Figure 3.9: 1:1 map overlaid on the physical environment. DARPA Distribution Statement "A" (Approved for Public Release, Distribution Unlimited).

Swarm Tactician-Rear STR is most likely located near the actual operation area, and the role is to deploy the mission created by SP to the swarm and oversee

its implementation. To suit this role, STR is initially presented with a similar miniaturized map interface as SP (Fig. 3.6), but STR also has the option to switch to a 1:1 map view that is similar to the STF perspective (Fig. 3.8 and 3.9). The ability to switch perspectives is relevant to STR if they relocate to within the operation area and require an egocentric view of swarm activity. STR can view or edit the details of an existing tactic list and add new tactics. After the tactics are finalized, the STR deploys them to the swarm by triggering the “Send to Swarm” button located at the bottom-right corner of the tactic list window in the table-top map view (Fig. 3.6). This disseminates the tactics to the swarm and initiates the consensus decision algorithm.

We also assumed that the STR role is responsible for making decisions on immediate events that may occur during operation. This was represented by the autonomous creation of a “Surveillance” tactic by one of the drones when it detects a mobile robot that is not a member of the swarm. When this occurs, five drones start following the moving target ((Fig. 3.10)) and later a notification message is sent to STR, which queries STR to either disable or ignore the detected mobile robot (Fig. 3.11).

Sever-Client communication

Sever-Client communication development used two packages in Unity: Photon Unity Networking (PUN) and Mixed Reality Toolkit (MRTK). PUN is a Unity package mainly for multiplayer online games, and it allows users to create/join a room either in local or cloud where Unity game objects can be synced over the network. MRTK is also a Unity package developed by Microsoft, and it is a toolkit for building custom MR applications with the Unity3D game engine. We use PUN to transmit tactic data across Server/Clients and sync user’s and swarms’ information (Fig. 3.2). MRTK provides many services for developers to add their custom methods, the most important one for our project is extension service. It allows the developer to create

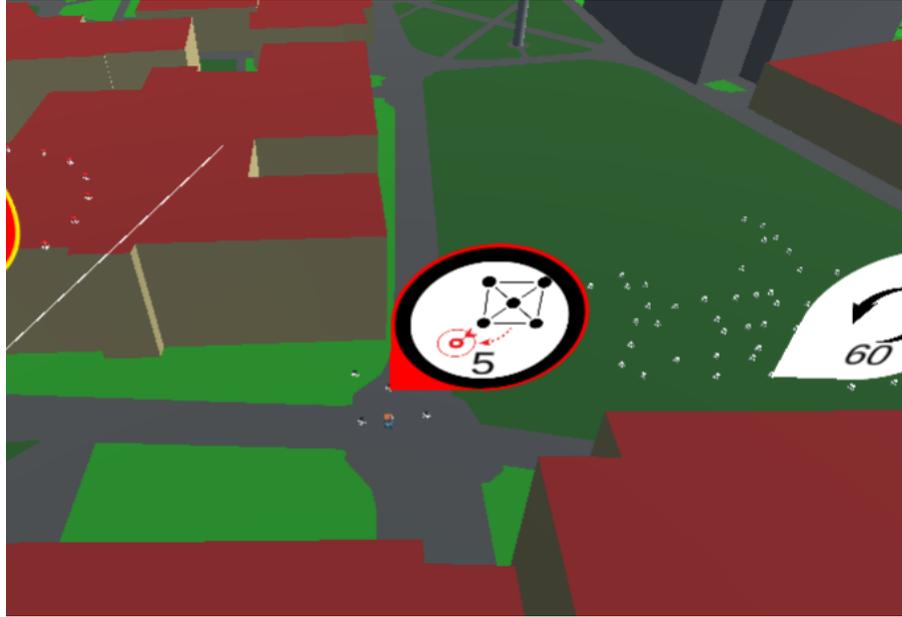


Figure 3.10: A miniaturized map view of 5 drones implement a “Surveillance” tactic for STR. DARPA Distribution Statement ”A” (Approved for Public Release, Distribution Unlimited).

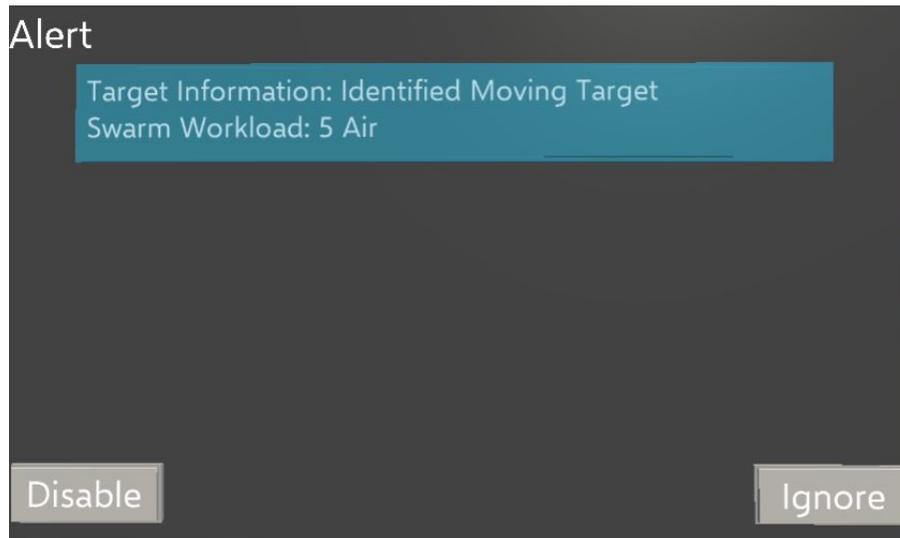


Figure 3.11: A notification alert menu is waiting for STR’s decision on moving target. DARPA Distribution Statement ”A” (Approved for Public Release, Distribution Unlimited).

their own functionalities. We developed several extension services, some of the essential services will be described below. “RoleService” can get/set the selected role in the main scene to define all the elements related to different personas. “RepositorySer-

vice” facilitates the Photon Networking level information exchange for synchronizing data repositories. The data repository system is a set of classes that allow in-memory storage and modification of collections of data. “TaskPlanningService” is the central controller for tactic planning. It can perform many operations (Add tasks, update tasks, delete tasks, and send task list to swarm, etc.) between user interface menu and data repository. “SwarmListService” acts similar to “TaskPlanningService”, it controls operations between Swarms information and data repository. Combining PUN and MRTK’s extension service, the Server-Client communication allows users to retrieve data from repositories and update the corresponding repository.

Client-ROS communication

The reason to establish Client-ROS communication is for SP and STR to have an alternative way to perform selecting operation with processed EMG signal and receive tactile feedback other than visual & sound feedback through ROS (Fig. 3.2). To accomplish these functionalities, a Unity library called ROSBridgeLib provides the ability to establish communication between our HoloLens Client and ROS side through RosBridge. RosBridge provides a JSON API to ROS functionality for non-ROS programs. To decode the processed EMG from ROS, a “HandDirectionSubscriber” was used to receive hand direction EMG data (classified using algorithms from [38, 39, 40]) as a JointState message. After the system receives and deserializes the message, a selection operation will be registered if the reads are matching the predefined selection input requirements (time and threshold). Another Function named “BuzzerPublisher” gets called whenever the Client needs to trigger the tactile feedback for users through either vibrotactile motor glove (controlled by ROS side) or sensory stimulation of the implanted nerve cuff electrodes (data transferred through ROS). There are two different levels of intensity of the tactile feedback. A normal intensity level of the feedback is for confirming the tactic list has been received from

the Server side. Higher intensity haptic feedback only happens when a notification alert message is received on the STR side. The system will send a serialized String message to ROS based on what type of tactile feedback is needed.

3.3 DARPA OFFSET Field Exercise 3

This HST system controlled by Air Tap was integrated with telemetry from a heterogeneous swarm consisting of 20 aerial and 20 mobile robotic agents and demonstrated at Camp Shelby, MS as part of an OFFSET Field Exercise 3. The 3D map of the exercise site was provided to us by DARPA and generated by an aerial drone flyover. We successfully visualized real-time swarm telemetry data that was generated from GPS coordinates of individual robotic swarm agents. Visualizations were demonstrated for both miniature map and first-person views. Figure 3.12 shows the bird’s-eye (top video frame) and first-person (bottom video frame) viewpoints of our mixed-reality interface showing the swarm’s mobile agents relative to a 3D map of FX3. Users can see robotic agents’ positions as relayed by live CCAST telemetry in the context of a miniature map view or look toward the veridical locations of each swarm unit and see their computer-generated positions through the building walls with our semi-transparent 1:1 map model. This demonstrates the integration of CCAST live telemetry with our Sprint 3 mixed reality interface.

3.4 Experiment Procedure

To evaluate the task performance of using different human inputs with the MR user interface participants performed the tactic planning process as the SP role (Fig. 3.7). Participants need to plan all three tactics: “Aggregation”, “Dispersion”, and “Secure Boundary” with the different types of input methods for three trials. Input methods include HoloLens’ “Air Tap” gesture, skin surface-EMG, and implant-EMG control.



(a)



(b)

Figure 3.12: Swarm hardware telemetry visualized on miniature map view for SP and STR (a) and first person view for STF (b) demonstrated at DARPA OFFSET Field Exercise 3. Both views are visualizing the same ground-based swarm agents and regions of the exercise site. In the first-person view, computer-generated locations of the swarm agents and outlines of the site buildings can be seen through an exterior wall. DARPA Distribution Statement "A" (Approved for Public Release, Distribution Unlimited).

The time of the planning procedure was recorded, and the participant was asked how they feel about the experience with the interface at the end of all three trials.

3.5 Participant

Due to the limitation of project duration and the pandemic, we were only able to do the experiment with 1 participant with amputation. The participant was asked to perform the tactic planning process for three trials with 15 minutes break in between. Instruction of procedure was presented to the participant during the experiment.

3.6 Results and Discussion

For the participant with amputation, the average time to complete the tactic planning process was 2 minutes 53 seconds. However, the third trial only took the participant 2 minutes 32 seconds to complete with the first (3 minutes 20 seconds) and second (2 minutes 48 seconds) trials' practices. The participant's completion times were still comparable to our able-bodied study staffs' (Fig. 3.13), whose times ranged from 1 minute and 41 seconds (using HoloLens' "Air Tap" gesture control) to 2 minutes and 11 seconds (using skin surface-EMG control).

After the trials, several user experience questions were presented to the participant for any thoughts of the interface. The participant felt the MR headset less disorienting than his other VR headset experience and also allowed him to easily walk around in the physical environment while getting a good view of the miniaturized map. The participant also felt that the trial got easier when he concentrated on the virtual environment. Additionally, the participant liked the tactile feedback, which provided a good confirmation when he selected the "Send to STR" button, and he wanted more confirmation feedback for other selection operations. Overall, the participant believed that the interface was "a pretty usable, easy system to use" and that "as much as I have played with a mouse in my life or whatever, um, I would probably enjoy this more. It is more interactive, especially the feedback." The participant's feedback was consistent with some of the feedback collected from our own study staff

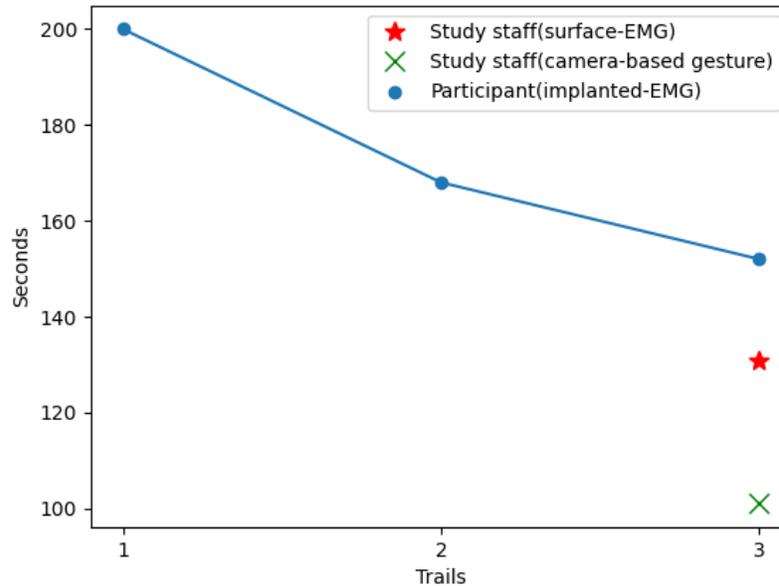


Figure 3.13: The completion times of tactic planning process for our amputee participant and study staffs. DARPA Distribution Statement "A" (Approved for Public Release, Distribution Unlimited).

as a part of the development process. Specifically, that the using EMG "as a button and the HoloLens as a display was a natural way to manipulate the environment" and that "the system is easy to become familiar with and develop a routine for".

From these results, we demonstrated that our implant-EMG user input control for amputee users could be used in an intuitive way and at a comparable rate to able-bodied users. Also, the combination of HoloLens device and implant-EMG control allows amputee users to interact MR interface without needing a hand, which can possibly expand to many different user interface designs where allows amputee users to interact with any interface that generally required the use of the hand easily. Additionally, tactile feedback can be an intuitive way to inform users without additional steps to look the information up.

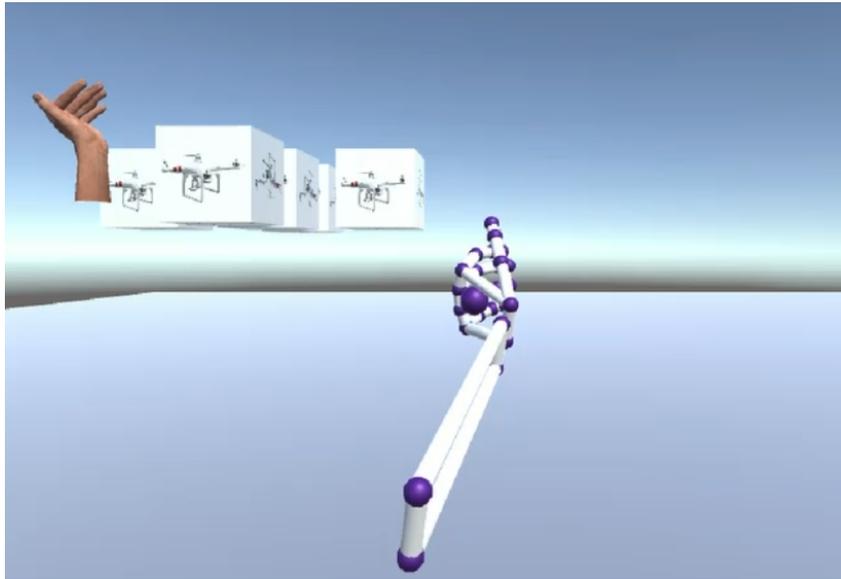
3.7 Future Work

Some opportunities for improving the system were identified during our development that should be addressed in future work. Specifically, the skin surface EMG acquisition methods need to be revised to be suitable for independent donning/doffing and field use. Wireless EMG acquisition systems exist and may be suitable for future versions of our HST system. Although rare in the literature [63], one of our study staff experienced motion sickness using HoloLens. Therefore, future experiments will need to screen for and exclude individuals who experience motion sickness using HoloLens.

The current work is only a proof-of-concept that was demonstrated by study staff and one case study of a person with below-elbow amputation, so additional experiments on both naive able-bodied and amputee participants are needed to determine generalizable results for task completion times and factors affecting HST (vigilance, situational awareness, workload, and human error).

Additionally, the current STF role was constrained to observing swarm activities without direct swarm interaction. However, an initial concept of using hand gestures to control drones' movements was implemented. The pilot version of direct swarm control with hand gesture using Leap Motion (Ultraleap, Ltd. Bristol, UK.) to move a swarm of simulated drones is shown in Figure 3.14 a, and the MR user interface shown EMG-controlled hand avatar for direct swarm control is shown in Figure 3.14 b. These will enable the direct control ability for both able-bodied and amputee users. In this direct control mode, each of the 3 degrees of hand motion controls an axis of drone movement. In the neutral position (palm perpendicular to the ground, wrist straight, and fingers at a 90 degree angle to the palm) drones do not move. All drone motions occur at a velocity proportional to the degree of hand motion, so the hand must return to neutral position to stop drone motion. An example with left hand, finger extension moves the drones forward at a velocity proportional to the angle of finger extension. Finger flexion moves the drones backward at a velocity

proportional to the angle of finger flexion. Wrist extension moves the drones laterally to the left and flexion moves them laterally to the right. Wrist supination increases their elevation and wrist pronation decreases it. The user's head rotation about the azimuth controls the heading orientation. Head rotation the right rotates drone orientation to the right. However, this functionality was not included in the current interface or experiment due to time and human participant safety constraints related to DARPA Field Exercise #3. Therefore, a direct swarm control subsystem can be fully implemented into the system in future work.



(a) An example of left hand's wrist extension moves a swarm of simulated drones to the laterally left.



(b) A grid-like hand avatar in HoloLens UI that is controlled by EMG signal.

Figure 3.14: Conceptual direct swarm control. DARPA Distribution Statement "A" (Approved for Public Release, Distribution Unlimited).

Chapter 4

Conclusions

This dissertation focused on evaluating human hand-reaching motor performance in MR and VR, as well as the development of a multi-user mixed reality interface for human-swarm collaboration.

The first part of this dissertation was to better understand the 3D point-to-point reaching performance in both VR and MR environments. This study performed six different performance measures for two types of point-to-point reaching tasks (finger pointing and cup placement tasks) with three environment conditions based on Fitts' law. A key finding of this study was that there is a trade-off between reaching motion confidence and accuracy. Specifically, participants tended to move fastest and smoothest in the MR without finger or cup indicator conditions, but their endpoint error was also highest. Additionally, in conditions where an indicator was provided in MR, finger pointing and cup placement task performances were comparable to the VR condition. These results are important because they suggest that MR tasks requiring low endpoint error require an endpoint indicator to be provided in the user interface. However, if endpoint error is not critical to the application, then not visualizing the endpoint can improve users' movement speed and smoothness.

The second part of this dissertation described an innovative multi-users human-

swarm teaming system using immersive MR technology. This system integrated the multi-modal MR visualization with other novel technologies including the consensus decision making algorithm, implanted/skin-surface EMG input, and tactile feedback. Our MR user interface was designed to meet the needs of more simultaneous users and scale to large drone swarms. Additionally, we have shown that our system can be integrated with real robotic swarm at DARPA OFFSET Field Exercise 3. We also showed a proof-of-concept that this system can also be adapted for a person with upper limb amputation to perform HST tasks where the performance is comparable to able-bodied users. However, due to the limitation of sample size and the responses from our participant, our system needs to be further refined with more experimental evaluations.

Appendix A

Questionnaire

The seven-point likert scale questionnaire for the participants' visual impression of our mixed reality device is shown below. Questions from 6 to 12 are responsible for participants' visual impression.

1. How familiar are you with video games?
2. How familiar are you with virtual reality?
3. How familiar are you with mixed reality/ augmented reality?
4. How comfortable the HoloLens was?
5. whether the clicked audio response was helpful?
6. Was watching the virtual objects just as natural as watching the real world?
7. Did you have the impression that the virtual objects belonged to the real object (finger cursor or cup), or did they seem separate from it?
8. Did you have the impression that you could have touched and grasped the virtual objects?
9. Did the virtual objects appear to be (visualized) on a screen, or did you have the impression that they were located in space?
10. Did you have the impression of seeing the virtual objects as merely flat images or as three-dimensional objects?

11. Did you pay attention at all to the difference between real and virtual objects?
12. Did you have to make an effort to recognize the virtual objects as being three-dimensional?
13. Would you like to try a similar technology again?

Bibliography

- [1] P. M. Fitts, “The information capacity of the human motor system in controlling the amplitude of movement,” *Journal of Experimental Psychology*, vol. 47, no. 6, p. 381, 1954.
- [2] P. M. Fitts and J. R. Peterson, “Information capacity of discrete motor responses,” *Journal of Experimental Psychology*, vol. 67, no. 2, p. 103, 1964.
- [3] I. MacKenzie and W. Buxton, “Extending fitts’ law to two-dimensional tasks,” 01 1992, pp. 219–226.
- [4] A. Murata and H. Iwase, “Extending fitts’ law to a three-dimensional pointing task,” *Human Movement Science*, vol. 20, no. 6, pp. 791 – 805, 2001.
- [5] M. Fu, A. Hershberger, K. Sano, and M. Çavuşoğlu, “Effect of visuo-motor co-location on 3d fitts’ task performance in physical and virtual environments,” *Teleoperators and Virtual Environments - Presence*, vol. 21, pp. 305–320, 08 2012.
- [6] M. Khademi, H. Mousavi Hondori, L. Dodakian, S. Cramer, and C. Lopes, “Comparing ‘pick and place’ task in spatial augmented reality versus non-immersive virtual reality for rehabilitation setting,” *Conference proceedings : ... Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Conference*, vol. 2013, pp. 4613–4616, 07 2013.
- [7] A. Hussein, L. Ghignone, T. Nguyen, N. Salimi, H. Nguyen, M. Wang, and H. A. Abbass, “Towards Bi-Directional Communication in Human-Swarm Teaming: A Survey,” *arXiv e-prints*, p. arXiv:1803.03093, Mar 2018.
- [8] P. Chaffey, R. Artstein, K. Georgila, K. A. Pollard, S. N. Gilani, D. Krum, D. Nelson, K. Huynh, A. Gainer, S. Alavi, *et al.*, “Developing a virtual reality wildfire simulation to analyze human communication and interaction with a robotic swarm during emergencies,” in *Proceedings of the 9th Language and Technology Conference*, 2019.
- [9] J. Wilbraham, “Command of aircraft swarms through a vr environment using gesture recognition,” *The UNSW Canberra at ADFA Journal of Undergraduate Engineering Research*, vol. 10, no. 1, 2018.

- [10] C. Zhao and M. J. Fu, "Evaluation point-to-point reaching performance in mixed reality and virtual reality," N.D., in preparation.
- [11] I. S. MacKenzie, "Fitts' law as a research and design tool in human-computer interaction," *Hum.-Comput. Interact.*, vol. 7, no. 1, pp. 91–139, Mar. 1992.
- [12] R. Soukoreff and I. MacKenzie, "Towards a standard for pointing device evaluation, perspectives on 27 years of fitts' law research in hci," *International Journal of Human-Computer Studies*, vol. 61, pp. 751–789, 12 2004.
- [13] T. Grossman and R. Balakrishnan, "Pointing at trivariate targets in 3d environments," *Conference on Human Factors in Computing Systems - Proceedings*, pp. 447–454, 01 2004.
- [14] Y. Cha and R. Myung, "Extended fitts' law for 3d pointing tasks using 3d target arrangements," *International Journal of Industrial Ergonomics*, vol. 43, p. 350–355, 07 2013.
- [15] R. Teather and W. Stuerzlinger, "Target pointing in 3d user interfaces," *CEUR Workshop Proceedings*, vol. 588, 01 2010.
- [16] I. E. Sutherland, "The ultimate display," in *Proceedings of the IFIP Congress*, 1965, pp. 506–508.
- [17] C. Ware, K. Arthur, and K. S. Booth, "Fish tank virtual reality," in *Proceedings of the INTERACT'93 and CHI'93 conference on Human factors in computing systems*, 1993, pp. 37–42.
- [18] C. Cruz-Neira, D. J. Sandin, and T. A. DeFanti, "Surround-screen projection-based virtual reality: the design and implementation of the cave," in *Proceedings of the 20th annual conference on Computer graphics and interactive techniques*, 1993, pp. 135–142.
- [19] I. E. Sutherland, "A head-mounted three dimensional display," in *Proceedings of the December 9-11, 1968, fall joint computer conference, part I*, 1968, pp. 757–764.
- [20] B. Sousa Santos, P. Dias, A. Pimentel, J.-W. Baggerman, C. Ferreira, S. Silva, and J. Madeira, "Head-mounted display versus desktop for 3d navigation in virtual reality: a user study," *Multimedia Tools and Applications*, vol. 41, no. 1, p. 161, Aug 2008. [Online]. Available: <https://doi.org/10.1007/s11042-008-0223-2>
- [21] P. Milgram and F. Kishino, "A taxonomy of mixed reality visual displays," *IEICE Trans. Information Systems*, vol. vol. E77-D, no. 12, pp. 1321–1329, 12 1994.
- [22] L. Rosenberg, "The use of virtual fixtures as perceptual overlays to enhance operator performance in remote environments," p. 52, 09 1992.

- [23] Microsoft Corp. Hololens (1st gen) hardware. [Online]. Available: <https://docs.microsoft.com/en-us/hololens/hololens1-hardware>
- [24] C. S. Mang, K. L. Campbell, C. J. Ross, and L. A. Boyd, “Promoting Neuroplasticity for Motor Rehabilitation After Stroke: Considering the Effects of Aerobic Exercise and Genetic Variation on Brain-Derived Neurotrophic Factor,” *Physical Therapy*, vol. 93, no. 12, pp. 1707–1716, 12 2013.
- [25] E. Scalona, D. Hayes, Z. Del Prete, E. Palermo, and S. Rossi, “Perturbed point-to-point reaching tasks in a 3d environment using a portable haptic device,” *Electronics*, vol. 8, no. 1, p. 32, 2019.
- [26] M. K. Holden, “Virtual environments for motor rehabilitation: Review,” *Cyberpsychology Behavior and Social Networking*, vol. 8, no. 3, pp. 187–211, 2005.
- [27] T. Pridmore, B. (Hons, S. Cobb, D. Hilton, J. Green, and R. Eastgate, “Mixed reality environments in stroke rehabilitation: Interfaces across the real/virtual divide,” *Virtual Reality & Assoc. Tech*, vol. 6, 01 2004.
- [28] M. Duff, Y. Chen, S. Attygalle, J. Herman, H. Sundaram, G. Qian, J. He, and T. Rikakis, “An adaptive mixed reality training system for stroke rehabilitation,” *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 18, no. 5, pp. 531–541, Oct 2010.
- [29] T. Ha and W. Woo, “An empirical evaluation of virtual hand techniques for 3d object manipulation in a tangible augmented reality environment,” in *2010 IEEE Symposium on 3D User Interfaces (3DUI)*. IEEE, 2010, pp. 91–98.
- [30] C. SV, T. Starmer, S. Tymms, T. Pridmore, and D. Webster, “Interfacing tangible input devices to a 3d virtual environment for users with special needs,” *Usability Evaluation and Interface Design: Cognitive Engineering, Intelligent Agents, and Virtual Reality*, vol. 1, p. 36, 2001.
- [31] H. M. Hondori, M. Khademi, A. McKenzie, L. Dodakian, C. V. Lopes, and S. C. Cramer, “Utility of augmented reality in relation to virtual reality in stroke rehabilitation,” *Stroke*, vol. 45, p. 1, 2014.
- [32] T. T. Blackmon, M. C. Cavusoglu, Fuji Lai, and L. W. Stark, “Human hand trajectory analysis in point-and-direct telerobotics,” in *1997 8th International Conference on Advanced Robotics. Proceedings. ICAR’97*, 1997, pp. 927–932.
- [33] S. Zhai and P. Milgram, “Quantifying coordination in multiple dof movement and its application to evaluating 6 dof input devices,” in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ser. CHI ’98. USA: ACM Press/Addison-Wesley Publishing Co., 1998, p. 320–327. [Online]. Available: <https://doi.org/10.1145/274644.274689>
- [34] H. Regenbrecht and T. Schubert, “Measuring presence in augmented reality environments: design and a first test of a questionnaire,” *Porto, Portugal*, 2002.

- [35] DARPA, *Teams Test Swarm Autonomy in Second Major OFFSET Field Experiment*, August 7, 2019. [Online]. Available: <https://www.darpa.mil/news-events/2019-08-07>
- [36] Y. Liu and K. Lee, “Probabilistic consensus decision making algorithm for artificial swarm of primitive robots,” *SN Applied Sciences*, vol. 2, 01 2020.
- [37] Y. Liu, “Distributed algorithms for consensus and formation control in scalable robotic swarms,” Ph.D. dissertation, Case Western Reserve University, 2018.
- [38] E. L. Graczyk, M. A. Schiefer, H. P. Saal, B. P. Delhaye, S. J. Bensmaia, and D. J. Tyler, “The neural basis of perceived intensity in natural and artificial touch,” *Science translational medicine*, vol. 8, no. 362, pp. 362ra142–362ra142, 2016.
- [39] D. W. Tan, M. A. Schiefer, M. W. Keith, J. R. Anderson, and D. J. Tyler, “Stability and selectivity of a chronic, multi-contact cuff electrode for sensory stimulation in human amputees,” *Journal of neural engineering*, vol. 12, no. 2, p. 026002, 2015.
- [40] M. Schiefer, D. Tan, S. M. Sidek, and D. J. Tyler, “Sensory feedback by peripheral nerve stimulation improves task performance in individuals with upper limb loss using a myoelectric prosthesis,” *Journal of neural engineering*, vol. 13, no. 1, p. 016001, 2015.
- [41] C. Zhao, C. Zheng, L. Roldan, T. Shkurti, A. Nahari, W. Newman, D. Tyler, K. Lee, and M. Fu, “Adaptive mixed-reality sensorimotor interface for human-swarm teaming simulation in field exercise,” N.D., in preparation.
- [42] J. Scholtz, “Theory and evaluation of human robot interactions,” in *36th Annual Hawaii International Conference on System Sciences, 2003. Proceedings of the IEEE*, 2003, pp. 10–pp.
- [43] A. Hussein and H. Abbass, “Mixed initiative systems for human-swarm interaction: Opportunities and challenges,” in *2018 2nd Annual Systems Modelling Conference (SMC)*. IEEE, 2018, pp. 1–8.
- [44] J. Y. Chen and M. J. Barnes, “Human–agent teaming for multirobot control: A review of human factors issues,” *IEEE Transactions on Human-Machine Systems*, vol. 44, no. 1, pp. 13–29, 2014.
- [45] J. A. Adams, “Critical considerations for human-robot interface development,” in *Proceedings of 2002 AAAI Fall Symposium*, 2002, pp. 1–8.
- [46] S. Bashyal and G. K. Venayagamoorthy, “Human swarm interaction for radiation source search and localization,” in *2008 IEEE Swarm Intelligence Symposium*. IEEE, 2008, pp. 1–8.

- [47] P. Chaffey, R. Artstein, K. Georgila, K. A. Pollard, S. N. Gilani, D. M. Krum, D. Nelson, K. Huynh, A. Gainer, S. H. Alavi, *et al.*, “Human swarm interaction using plays, audibles, and a virtual spokesperson,” in *Artificial Intelligence and Machine Learning for Multi-Domain Operations Applications II*, vol. 11413. International Society for Optics and Photonics, 2020, p. 114130V.
- [48] A. M. Naghsh, J. Gancet, A. Tanoto, and C. Roast, “Analysis and design of human-robot swarm interaction in firefighting,” in *RO-MAN 2008-The 17th IEEE International Symposium on Robot and Human Interactive Communication*. IEEE, 2008, pp. 255–260.
- [49] J. Nagi, A. Giusti, L. M. Gambardella, and G. A. Di Caro, “Human-swarm interaction using spatial gestures,” in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2014, pp. 3834–3841.
- [50] A. Rogers, “A usability perspective into gesture for human-swarm teaming,” *The UNSW Canberra at ADFA Journal of Undergraduate Engineering Research*, vol. 9, no. 1, 2018.
- [51] A. Couture-Beil, R. T. Vaughan, and G. Mori, “Selecting and commanding individual robots in a multi-robot system,” in *2010 Canadian Conference on Computer and Robot Vision*. IEEE, 2010, pp. 159–166.
- [52] M. Chandarana, E. L. Meszaros, A. Trujillo, and B. D. Allen, “‘fly like this’: Natural language interface for uav mission planning,” 2017.
- [53] M. Chandarana, A. Trujillo, K. Shimada, and B. D. Allen, “A natural interaction interface for uavs using intuitive gesture recognition,” in *Advances in Human Factors in Robots and Unmanned Systems*. Springer, 2017, pp. 387–398.
- [54] R. Ibrahimov, E. Tsykunov, V. Shirokun, A. Somov, and D. Tsetserukou, “Dronepick: Object picking and delivery teleoperation with the drone controlled by a wearable tactile display,” in *2019 28th IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*. IEEE, 2019, pp. 1–6.
- [55] E. Tsykunov, R. Agishev, R. Ibrahimov, L. Labazanova, A. Tleugazy, and D. Tsetserukou, “Swarmtouch: Guiding a swarm of micro-quadrotors with impedance control using a wearable tactile interface,” *IEEE transactions on haptics*, vol. 12, no. 3, pp. 363–374, 2019.
- [56] N. Anil and S. Sreeletha, “Emg based gesture recognition using machine learning,” in *2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS)*. IEEE, 2018, pp. 1560–1564.
- [57] D. W. Tan, M. A. Schiefer, M. W. Keith, J. R. Anderson, J. Tyler, and D. J. Tyler, “A neural interface provides long-term stable natural touch perception,” *Science translational medicine*, vol. 6, no. 257, pp. 257ra138–257ra138, 2014.

- [58] K. S. Haring, V. Finomore, D. Muramoto, N. L. Tenhundfeld, J. Wen, and B. Tidball, “Analysis of using virtual reality (vr) for command and control applications of multi-robot systems,” in *Proceedings of the 1st International Workshop on Virtual, Augmented, and Mixed Reality for HRI (VAM-HRI)*, 2018.
- [59] O. Erat, W. A. Isop, D. Kalkofen, and D. Schmalstieg, “Drone-augmented human vision: Exocentric control for drones exploring hidden areas,” *IEEE transactions on visualization and computer graphics*, vol. 24, no. 4, pp. 1437–1446, 2018.
- [60] J. Ruiz, A. Viguria, J. Martinez-de Dios, and A. Ollero, “Immersive displays for building spatial knowledge in multi-uav operations,” in *2015 International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE, 2015, pp. 1043–1048.
- [61] M. Daily, Y. Cho, K. Martin, and D. Payton, “World embedded interfaces for human-robot interaction,” in *36th Annual Hawaii International Conference on System Sciences, 2003. Proceedings of the*. IEEE, 2003, pp. 6–pp.
- [62] J. Patel, Y. Xu, and C. Pinciroli, “Mixed-granularity human-swarm interaction,” in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 1059–1065.
- [63] A. Vovk, F. Wild, W. Guest, and T. Kuula, “Simulator Sickness in Augmented Reality Training Using the Microsoft HoloLens,” in *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems - CHI '18*. Montreal QC, Canada: ACM Press, 2018, pp. 1–9. [Online]. Available: <http://dl.acm.org/citation.cfm?doid=3173574.3173783>