Doctoral Dissertation

User Interaction for Handheld Augmented Reality in Task Support

Varunyu Fuvattanasilp

March 17, 2021

Graduate School of Information Science
Nara Institute of Science and Technology

A Doctoral Dissertation
submitted to Graduate School of Information Science,
Nara Institute of Science and Technology
in partial fulfillment of the requirements for the degree of
Doctor of ENGINEERING

Varunyu Fuvattanasilp

Thesis Committee:

Professor Hirokazu Kato	(Supervisor)
Professor Kiyoshi Kiyokawa	(Co-supervisor)
Associate Professor Masayuki Kanbara	(Co-supervisor)
Assistant Professor Yuichiro Fujimoto	(Co-supervisor)
Doctor Alexander Plopski	(Co-supervisor)

User Interaction for Handheld Augmented Reality in Task Support[*]

Varunyu Fuvattanasilp

Abstract

In this thesis, I investigate the aspect of user interaction in Augmented reality for task support in a handheld device. I focus on a handheld device as it becomes a promising platform for many AR applications due to accessibility and popularity nowadays. I divided the user interaction into two groups based on role in task supporting: 1) Input the virtual instruction by an expert user, and 2) Receive and follow the virtual instruction by a novice user. In Chapter 2, I address the problem of an expert user for inputting the virtual instruction. In the task support scenario, where the object placement and manipulation required to perform. Thus, an intuitive object manipulation technique is needed for this task. I present SlidAR+, a Gravity-Aware 3D Object Manipulation for Handheld Augmented Reality. SlidAR+ is a method for controlling the position and orientation of virtual objects in HAR. Next, I present the results of experiments by comparing SlidAR+ and a state-of-the-art method to evaluate the performance of SlidAR+. In Chapter 3, I investigate the effect of the latency of the handheld device's camera on task performance. As for the novice user, following the instruction correctly is fundamental. However, looking through a camera lens in the video-see through displays affects users' performance due to distortions in visual representation and hardware performance. The effect of latency has not much been explored. To address this, I investigate the effect of the latency of the mobile phone's camera on task performance. I conducted two small studies: 1) To see which levels of latency users start to notice, and 2) How does latency affect the

[*] Doctoral Dissertation, Graduate School of Information Science, Nara Institute of Science and Technology, March 17, 2021.

task performance. To summarize in this thesis, I present two works: 1) SlidAR+: Gravity-Aware 3D Object Manipulation for Handheld Augmented Reality and 2) an investigation of the effect of latency on 2D display for micro-task.

Keywords:

Augmented Reality, Handheld device, Latency, Object Manipulation

Contents

List of Figures

List of Tables

1. Introduction

1.1 Augmented Reality

Augmented Reality (AR) is a technology that combine real and virtual world by enhance our physical world perception with the computer-generated information. This computer-generated information can be visual, auditory, haptic or even somatosensory and olfactory. Azuma et al.[2] define one of the most commonly accepted definitions of AR technology that it consisted of three characteristics:

- Combines real and virtual content
- Interactive in real-time
- Registration of virtual object in 3D

These three characteristics define the main requirements of the AR system, that the system has to combine real-world and virtual content and presented on the same screen. The virtual content must interact and respond to the user input in real-time. Finally, the system can track and enable the virtual content to place fixed in the real-world.

Milgram et al.[64] introduce an alternative way of defining AR as a part of the ”Mixed Reality” concept. A realty-virtuality continuum (Figure 1) which is a taxonomy of mixed reality concept combined virtual and real elements and divided it into four parts: real environment, Augmented Reality(AR), Augmented Virtuality(AV), and virtual environment. AR defined as where the virtual content used to enhance the user's view while the majority of view is the real world. AV is the opposite if AR, where most of the user's view is the virtual environment with only a small part of the real-world element.

1.2 Handheld Augmented Reality

Handheld augmented reality of HAR usually refer as AR application operates on handheld devices or displays such as smartphones, tablet computers, and other types of devices that can be carried and move while operated. Most of the handheld device comes with their operating system and uses a video see-through

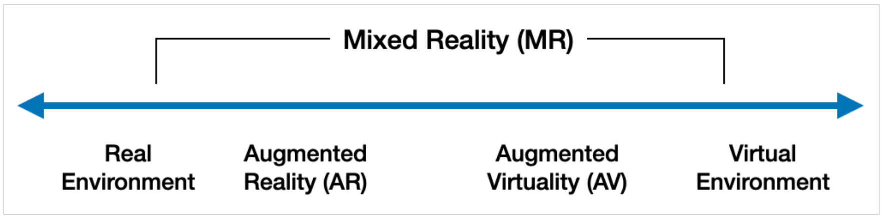


Figure 1: Milgram et al. [64] reality-virtuality continuum.

type of display with a touch screen as a main source of input. A handheld optical see-through display [95] and a mobile projector [86] also consider as a handheld device. However, usability and availability when we talk about a handheld device it mostly refers to a handheld video see-through where the live video of the real world was captured and stream using the device's camera then display on the device's display (usually on the opposite side of the camera).

With the advancement of technology in the past decade, the handheld device becomes faster, smaller, and lighter. HAR technology has also been improving and evolving from a backpack computer system to a mobile phone (Figure 2). Nowadays, a handheld device becomes a promising and appealing platform to deploy and develops an AR application due to the mobility and hardware performance that allow the researcher to explore the usefulness of AR on many new area and aspect.

1.3 Augmented Reality for Task Support

AR is currently being utilized in many field such as education [49, 91], or in medical application [25, 48, 93]. Industrial Augmented reality (IAR) also increasing in popularity among research topics, especially using AR in task support such as maintenance, inspection,and remote collaboration [12, 17, 32, 36, 31].

The advantage of using AR in task support is it allows us to observe data or insert information directly on the physical object in the real environment through a handheld device or HMD. Provide the user with the necessary visual instruction and information which has proven to improve efficiency while performing the task [31, 38, 98].

There are many types of augmented reality for task support applications.

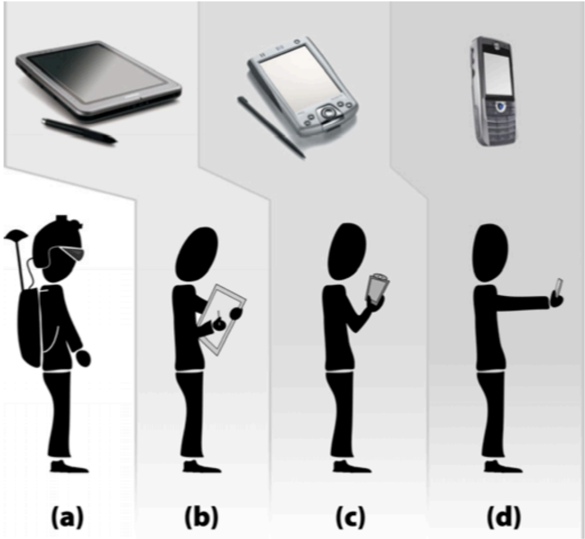


Figure 2: The evolution of mobile AR system to HAR: (a) A backpack with head mounted display, (b) Ultramobile personal computer (UMPC), (c) Personal digital assistant (PDAs), (d) Mobile phone [103].

We can generally divide it into two groups; Asynchronous and Synchronous task support.

1.3.1 Asynchronous Task Support

In the asynchronous task support system, the time of inputted and followed the virtual guidance happens separately. Generally, the virtual guidance was prepared before the task started. Many AR applications such as AR for maintenance, inspection, or observation can be considered as this type.

- **AR for Maintenance**

  AR has been developed to assist in maintenance, repair, and assembly task for a decade [100]. From a laser printer maintenance [28] to a wiring harness on an airplane's electrical system by Boeing [22, 65], many researchers and companies have worked in this area and several prototype systems have developed [52, 101, 79].

  This type of task usually have high complexity and requires well-trained users or a manual to perform. However, physical manual instruction with text and pictures might not provide enough information and difficult to understand for a task with a complex machine. AR allows instructions to be available as 3D drawing, annotation, or actual 3D model upon the working area, showing how to perform the task step-by-step that needs to be done.

- **AR for Inspection & Observation**
  Another use of AR in task support is in an inspection or observation task where a user has to observe and check the condition, status, and location of various targets in the working area. These tasks usually do not require physical manipulation or perform directly at the targets. For example, machines and pieces of equipment in the factory need a daily inspection to make sure that everything can work properly. In this situation a checklist that contains all information about targets that need to be checked is commonly used. Rather than carrying a physical checklist, AR technology can provide all of the checklist information through HMD [13, 72] or a handheld device [81]. Some utilize AR to provide book information while walking in the library [29, 85, 87] or for navigation [33].

1.3.2 Synchronous Task Support

In the synchronous task support system, the virtual guidance was inputted in real-time during the task. This system requires two or more users to operate at the same time. The most commonly known of this type is the AR remote collaboration.

- **AR Remote Collaboration**
  AR remote collaboration usually refers to a system that requires two users to working through an online video conference. One user is a person who stays at the workplace, can refer to as a local or novice user who asked for assistance or help. Another user is an expert who provides instruction and guidance through a video conference system from a faraway location. However, vocal guidance alone is not enough in a complex situation. With AR technology an expert user can input various types of computers generate

instruction directly into a novice user scene [8, 27, 31, 32, 57, 96, 102] in real-time.

1.4 HAR for task support

The advantage of using HAR in task support is that handheld devices are much more affordable than head-mounted display. Handheld also easier to input, can provide much information on the screen at the same time, larger field of view, high mobility, and longer operation time. The biggest disadvantage of handhelds is that it requires a user to hold the device to observe and see the AR content. This makes handheld does not suitable in a two hands operation task. Even with one hand operation can be difficult because the device can be an obstacle while performing the task. Holding the device with one hand for a long time can cause extra physical fatigue to an arm.

1.4.1 Interaction in task support

In this thesis, I focus on user interaction in task support using a handheld device. I divided an interaction into two group base on the user role (Figure 3):

- **Expert User: Input an instruction**

  The main role of an expert user is to observe the task, identify and diagnose the problem, create an instruction that can be in any form such as voice, annotation, 3D drawing, 3D model, or even a video. Then provide that information to the novice user. In case of an AR instruction the expert user has to register and place an instruction into the real world. The user of these systems can be an expert user, who prepared the instruction in an asynchronous task support or a remote user in a synchronous AR remote collaboration.

  Input an AR instruction is a fundamental interaction for an expert user. Placing guidance to the wrong position can cause an impact on task efficiency or even lead to failure. Most of the handheld devices use a touch screen method for inputting the information, it becomes difficult to input 3D information through a 2D input display. This problem can occur in both asynchronous and synchronous task support system. In an asynchronous system, the expert user can keep re-adjusting an instruction until it accurately placed in the correct position, but it still takes a lot of time and effort. This problem becomes more crucial in synchronous systems as the expert user does not have much time to re-adjust an instruction as they have to input it in real-time, but using voice communication can reduce this problem.

To overcome this problem and improve the usability of the expert user, an intuitive and efficient object manipulation method for a handheld device is needed to allow the expert user to place an AR guidance fast and correct.

- **Novice User: Follow an instruction**
  Novice user refers to a user who needs assistance or guidance while performing a task. The task can be the training process or on-site problem-solving. In AR task support the novice user observes and receives instruction provided by an expert user, then follow instructions to complete the task. The process of following AR instruction or transcribing digital information to the physical world is very important as it can affect a user's performance.

  Looking and performing through a see-through display such as handhelds can affect users' performance due to limitation of device that causes distortions in a visual representation such [19, 20] as distortion of binocular disparity, the introduction of geometric distortions, and system latency. These limitations affect user perception and the ability to transfer digital instruction to the task knowledge on both asynchronous and synchronous task support system. It is possible to overcome and reduce this problem in the synchronous system by using communication between expert and novice users. However, it can become a serious issue in the asynchronous system as the novice user has to rely on the instruction alone without an expert to correct their mistake.

  Thus, the studies to understand the effect of these limitations in the handheld device are essential if we want to develop a better task support system in a handheld device.

I consider these two user interactions essential to studies and understand the user behavior to design and develop a good task support system using HAR.

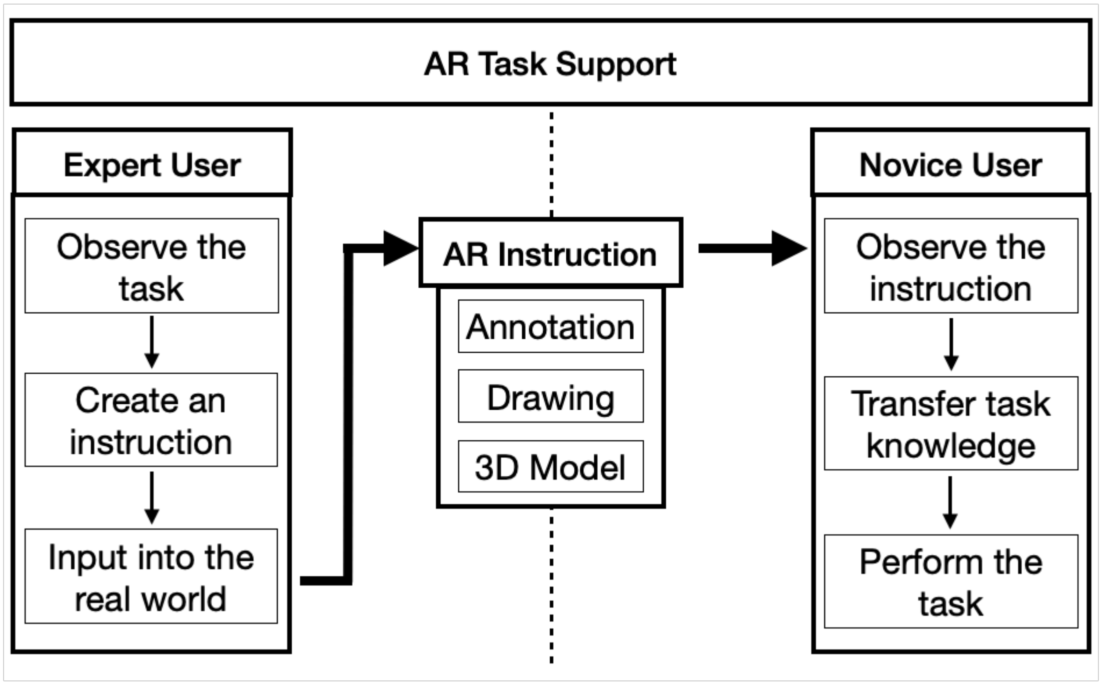


Figure 3: User interaction in task support based on their role.

1.5 Goal and Approach

In this thesis, I would like to focus on the asynchronous task support system as many problems for both expert and novice users can not be solved by real-time communication. I conducted two case studies for each role of user interaction in asynchronous task support using a handheld device. First, I introduce a 3D object manipulation technique for handheld augmented reality to improve the efficacy and reduce a mental load while placing an AR instruction. Second, I investigate the effect of latency on task performance.

1.5.1 Object manipulation in HAR

To place AR content into real-world, users have to control up to 6 degrees of freedom (DoFs) (3 DoFs for position, 3 DoFs for orientation), which is a difficult task as handheld devices are usually controlled through a 2D display. It is thus necessary to develop intuitive methods that allow users to control all 6 DoFs easily. Polvi et al. developed SlidAR [82], a 3D positioning technique for HAR this technique allows users to adjust and reposition virtual content by performing only a slide gesture. However, this technique can not be used for 3D object manipulation as it lacks the ability to control orientation. My goal is to extend SlidAR into fully 6 DoFs object manipulation techniques. My approach is to use the gravity information to assist in the rotation control. Then I created SlidAR+, a 6 DoFs object manipulation method in HAR, and conducted experiments to evaluate the efficacy of SlidAR+.

1.5.2 Effect of latency on the handheld device

Performing tasks while looking through a camera in the video-see through handheld display can affect users' performance due to the limitation of the monoscopic display that can causes distortions in a video output on display such as lack of depth perception, the geometric distortions, and system latency.

System latency or camera latency is a delay between an action and the display of the action on a device's screen in real-time. Latency can occur from the process of camera streaming to system processing before rendering an output video on the device's screen. Even with medical-grade equipment for surgery operation can have the camera latency up to 90 ms [53, 71]. Latency can be higher on commercial-grade handheld equipment as there are a lot of handheld devices with different hardware and software based on the manufacturer. This system latency does not even include the process of tracking, registration, and render of a virtual object in AR for task support scenarios.

This latency problem can affect tasks such as reduce accuracy and increase completion time [53]. Because it affects how the user responded to their action while performing the task. Especially in the HAR system where users have to perform tasks behind the device's camera and have to look through the device's display to see their hands. In some cases, they might move the device away or move to another viewpoint physically to see the actual environment. However by doing that the user has to keep switch between looking at the device and their hands. Several researchers try to overcome this problem by developing a near-zero latency system [23, 46]. However, this approach is expensive and nearly impossible to apply to the commercial handheld device.

My goal is to investigate the effect of camera latency on users following augmented reality instructions to understand and study the effect on user behavior and perception.

1.6 Contributions

The main contributions of this thesis are:

1. **SlidAR+**: 6DoF object manipulation for HAR

    - I present a novel 6 DoF objection manipulation method for handheld devices, by integrating the 3D rotation control using gravity with the position technique of SlidAR.
    - The insights gained from a user study that compares SlidAR+ with *Hybrid* to investigate the efficiency of using gravity information to pre-align and constrain the rotation of the virtual objects for 6 DoF tasks.

2. **Investigate the effect of camera latency on 2D display for micro-task**: I investigate the effect of camera latency in mobile phone/tablet's camera for small scale task that require high accuracy and precision. I conducted two studies to clarify this following points

    - The first study is to investigate what is the minimal latency the human can perceive.
    - The second study is to investigate how does latency affect time and accuracy when performing high precision task.

1.7 Thesis Structure

This thesis consists of four chapters (Figure 4). Chapter 1, I briefly explain the definition, background, goal, and contribution of my research. In Chapter 2, I introduce a SlidAR+: gravity Gravity-Aware 3D Object Manipulation for HAR including a literature review of object manipulation methods in HAR and evaluate SlidAR+ before present the results. In Chapter 3, I present the result of the effect of camera latency in task performing on a 2D display. I also include detail of the latency measurement system that I used. Finally, I conclude my results, finding, and possible future work in Chapter 4.

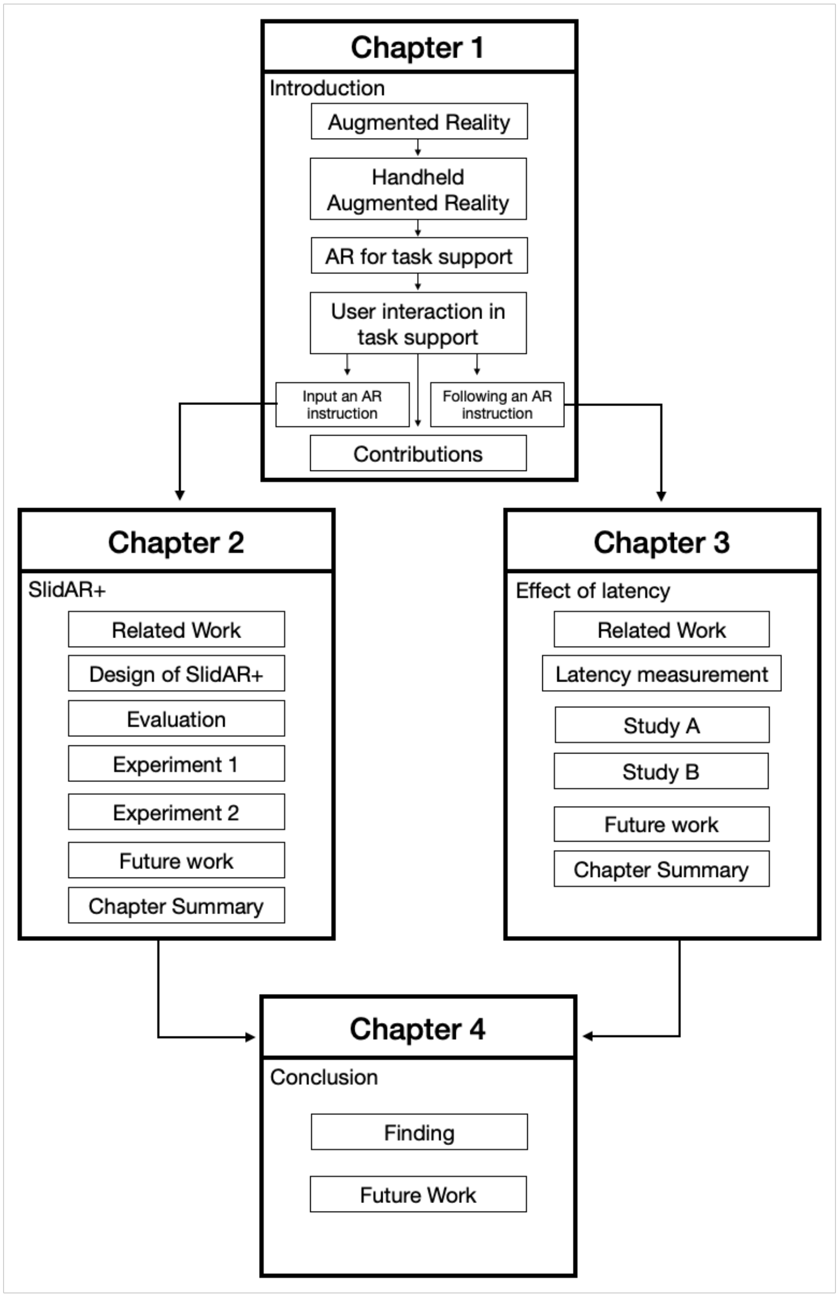


Figure 4: Thesis outline

2. SlidAR+: Gravity-Aware 3D Object Manipulation for Handheld Augmented Reality

Many HAR applications utilize markers for tracking [41, 66]. However, this severely limits an application's ability to be used more generally [11]. To address this, an increasing number of applications are utilizing visual simultaneous localization and mapping (vSLAM) [34, 51, 97] to track the pose of the device, without requiring users to set up fiducial markers or scan the environment beforehand.

Most HAR applications initially align objects with a world coordinate system. In general, the world coordinate system coincides with a fiducial marker, or is set via a vSLAM algorithm. In some cases, the orientation of the created object may not match the desired orientation and require manual adjustment by the user [14, 45, 61]. However, in order to manipulate a virtual object, users have to adjust up to 6 degrees of freedom (DoFs) (3 DoFs for position, 3 DoFs for orientation), which is a difficult task as handheld devices are usually controlled through a 2D display. It is thus necessary to develop intuitive methods that allow users to control all 6 DoFs easily.

Polvi et al. developed SlidAR [82], a 3D positioning technique for HAR application that utilizes ray casting and epipolar geometry to simplify the procedure of positioning a virtual object in the real world. SlidAR allows users to adjust and reposition virtual content by performing only a slide gesture. Polvi et al. compared SlidAR with a device-centric positioning method [82] and their results showed that users can position virtual content faster with SlidAR. However, this technique can not be used for 3D object manipulation as it lacks the ability to control orientation. This is because SlidAR was developed for a text-based AR annotation application that does not need orientation control. To place 3D AR content, users must be able to manipulate position as well as orientation of the object.

Bowman et al. [14], explained the benefit of using constraints in 3D user interfaces as they can simplify interaction while improving accuracy and user efficiency. Even though using constraints limits the amount of control the user can have, the simplification allows for a more intuitive interaction [1]. It can also reduce the number of DoFs that users have to control.

To address, I introduce SlidAR+, a method for controlling the rotation and position of virtual obtjecs in HAR applications. SlidAR+ extends SlidAR, with gravity-constrained rotation capabilities. This constraint is based on the observation that most common AR content will be placed relative to man-made objects in the environment that are mostly either horizontal or vertical. This observation has been previously utilized to improve the quality of surface tracking [54], interface design [74], and physically accurate rendering [55]. In SlidAR+, I first constrain the initialization of virtual objects to be either parallel or perpendicular to the gravity vector and then constrain one of the rotation axes to always align based on the gravity vector to allow for fast adjustments.

In this chapter, First I present an overview design of SlidAR+ and how to operate. Next, I present the evaluation of SlidAR+ include two user studies: the first one to confirm the performance of SlidAR and evaluate in positioning task. the second experiment, I evaluate a SlidAR+ in a task involves manipulating all 6 DoFs.

2.1 Related Work

Existing methods for object arrangement in HAR can be divided into: (1) methods that automatically align the virtual content with the physical world, and (2) methods that let users manually adjust the pose of the virtual content.

2.1.1 Automatic Alignment of AR content

Automatic alignment methods extract features from the environment to adjust the pose of virtual content, without any input from the user. Many methods align content with fiducial markers that are placed in the scene. The pose of the virtual object can thus be adjusted by adjusting the pose of the marker. In recent years, more and more methods have been developed for marker-less tracking. These systems use the surfaces of the detected environment to constrain and align pose

[1] https://developer.apple.com/design/human-interface-guidelines/ios/system-capabilities/augmented-reality/

of the virtual content [3, 54, 84, 89].

However, these methods depend heavily on the accuracy of surface detection and any errors can cause misalignment of the virtual object. In particular, if the target area has a complicated shape, image-based shape measurement techniques such as vSLAM may not be able to recover a surface accurately.

2.1.2 Manual Alignment if AR Cintent

To correct erroneous placement, users can manually adjust the pose of the virtual content. Over the past years, a large variety of techniques have been devised to simplify this process. In general, they can be sub-divided into four categories: button-based, screen-based, device-centric, and gesture-based.

Button-Based Manipulation

Button-based manipulation methods utilize physical and virtual buttons on the handheld devices to position and orient virtual objects. Herrysson et al. [39, 40] used a smartphone's physical buttons to control each DoF parameter with a distinct button. In a similar manner, Castle et al. [15] used virtual buttons on the device's display. These methods require at least two buttons to control 1 DoF parameter (one to increase and another to decrease). In all, they require 12 buttons to control the position and orientation of a virtual object. Bai et al. [5] combine button and screen-based manipulation wherein users freeze a frame and select the DoF they would like to control with virtual buttons located at the edge of each translation and rotation axis. Afterward, the parameter can be adjusted via finger scrolling on the screen.

Button-based methods can change only one parameter at a time, which takes a lot of time, needs a large number of buttons, and is difficult to operate when controlling multiple DoFs simultaneously.

Screen-Based Manipulation

Screen-based manipulation methods utilize hand or finger gestures while interacting directly with the screen to manipulate the pose of the virtual object. Most screen-based manipulation techniques share the same basic idea of assigning one type of gesture (e.g., a vertical or horizontal slide) to control one DoF parameter. This allows users to control more than one parameter at the same time. With ARCBALL [92], users can adjust the rotation of the object by sliding with a single finger into the direction they want to rotate the object. Similar ideas have been explored to control 6 DoFs with two-finger gestures [47, 56, 67, 88, 99]. Martinet et al. [59] developed the z-technique to control the depth of a virtual object wherein while one finger is touching the virtual object, the second finger moves horizontally across the screen to move the object farther from or closer to the user. They later expanded their work to control all 6 DoFs and depth with the Depth-Separated Screen-Space (DS3) method [60]. They used different types of gestures to control the direction and orientation, thereby minimizing any mistakes due to similar gestures. The Shallow-Depth 3D technique developed by Hancock et al. [37] extends DS3 to three-finger gestures to control all 6 DoFs. However, increasing the number of fingers to be used for gestures also increases the cognitive load demand and complexity of the interaction.

Screen-based manipulation methods are accurate and do not require the user to move the device. Many studies [39, 62, 67] have also shown that these methods are most suitable for rotation and scaling tasks, as the user is able to control these parameters very accurately. However, as all parameters are controlled on the screen, the increasing number of gestures users have to learn and the number of fingers involved in each gesture affect the intuitiveness and ease of manipulation.

Device-Centric Manipulation

Device-centric movement methods utilize the movement of handheld devices to control the position and orientation of the virtual object. By adjusting the 6 DoFs pose of the device, users can adjust the position and rotation of the virtual object simultaneously. To prevent unintentional adjustments, users can trigger when to start the adjustment. Henrysson et al. [39] developed a grasping technique where the user can manipulate a virtual object by continuously pressing the screen or a button while moving the device [39]. Mossel et al. [67] support the manipulation of the virtual object by highlighting its axes as virtual depth cues. Marzo et al. [62] combined a device-centric movement grasping technique with screen-based gestures to control position and orientation, respectively, to improve the accuracy and speed of object manipulation.

Device-centric methods have been found to be the fastest among the 4 types of methods discussed here. However, as they use the movement of the device, it is difficult to control individual parameters accurately [39, 40, 67, 73].

Gesture-Based Manipulation

Gesture-based manipulation methods use the device's camera to detect and track hand gestures performed in front of the camera to manipulate 6 DoFs of the virtual object.

Users can perform a variety of gestures, such as pushing, grabbing, or twisting [6, 18, 40, 44, 105] to manipulate the virtual object. Alternatively, these gestures can also be performed by other devices, like a pen [43].

Although users can control all 6 DoFs at the same time through gestures, these methods have been shown to be less effective than device-centric movement methods in practical scenarios [4, 40, 44].

Summary

Most previous studies focused on the efficiency with which users can control all 6 DoFs. On the other hand, I designed SlidAR+ to minimize the number of parameters that users have to control in order to adjust the pose of a virtual object. For positioning, users can place an object in the scene by adjusting only 1 DoF with SlidAR. For orientation, I utilize gravity information to constrain the initial pose of the virtual object to the most probable orientation (either parallel or perpendicular to the gravity vector), and to provide users with an option to perform gravity-constrained rotation. As previously discussed, most of planar surfaces in man-made are aligned either parallel or perpendicular to the gravity vector (wall, table, pillar, etc.). I made an assumption that in a general AR content placement task, the virtual contents are also likely to be placed aligned with the direction of gravity (parallel or perpendicular); hence, users would have to adjust only 1 DoF.

In that sense, SlidAR+ combines the features of automatic alignment during the initial placement of AR object phase with the ability to pre-align the virtual object to the physical world constraint and manipulability (screen-based). By using gravity information to control the pre-alignment, SlidAR+, allows the user to have more control over the initial pose. This can help to avoid the misalignment caused by the tracking system, which is a common issue in SLAM-based applications. Normally, the virtual object is set to align to the system coordinate. However, in SLAM-based tracking, it is hard to predict how the system coordinate will be initialized, i.e., whether the coordinate system will be aligned with the real world or not. In my approach, the initial pose is now fixed based on the direction of gravity instead of the system coordinate. Furthermore, SlidAR+ does not require any additional real-world sensing (such as using computer vision technique) or special hardware (such as a depth camera or a 3D construction of the environment).

2.2 Design of SlidAR+

SlidAR+ extends the capability of SlidAR [82] by adding the ability to orient virtual objects in HAR applications. I followed the original motivation behind SlidAR and aimed to reduce the number of DoFs that users have to control while manipulating a virtual object, especially when users want to place objects parallel or perpendicular to the gravity vector.

For the experiments discuss in Section 2.4 and 2.5, I implemented SlidAR+ in Unity3D[2] and used marker-based tracking from Vuforia SDK[3]. In a practical scenario, SlidAR+ can be used with marker-less tracking such as Apple ARKit[4] or Android ARCore[5] platforms. The current system simulates a task where the user has to place 3D AR models in the real environment and adjust their pose. I divide this task into two phases (Fig. 5): (1) initialization phase and (2) adjustment phase. The system also provides a set of 3D assets with an axis aligned to gravity.

2.2.1 Initial Placement of AR object

In the initial placement phase, a user first selects the desired type of annotation. After that, the user selects the alignment-type to determine whether the object

[2] https://unity.com/

[3] https://developer.vuforia.com/

[4] https://developer.apple.com/augmented-reality/

[5] https://developers.google.com/ar

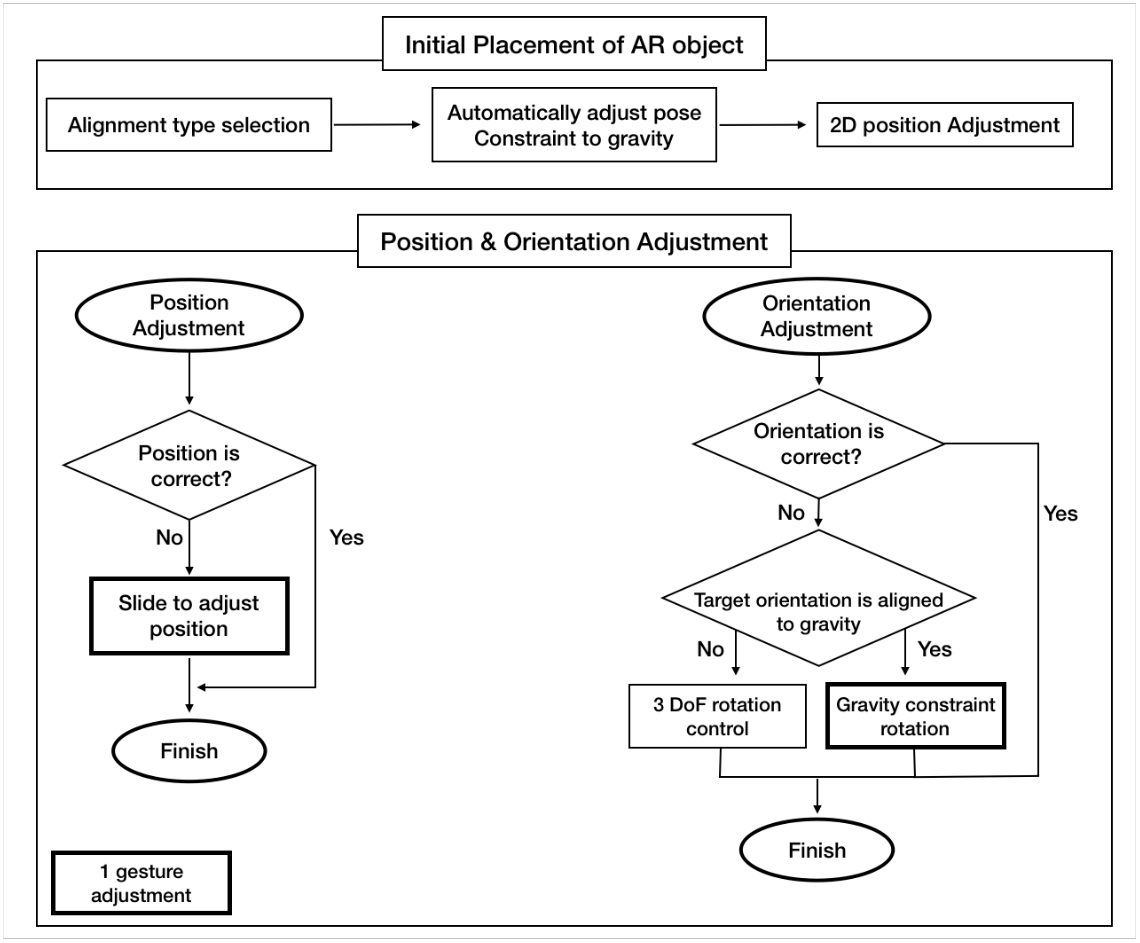


Figure 5: Two-phase workflow of SlidAR+ is divided into two phases: (1) In the initialization phase, users align the selected object with the direction of gravity. (2) During the adjustment phase, the position can be adjusted with SlidAR and the orientation can be corrected using gravity-constrained orientation control.

should be parallel or perpendicular to the direction of gravity and presses the desired location on the screen of the handheld device. The system places the virtual object at a predefined depth (fix distance from the front camera instead of using vSLAM to remove the effect of inaccuracy of vSLAM on the experimental results). Then aligns it with the desired direction according to the gravity information from the sensors built into the handheld device. During this phase, the user can adjust the position by pressing or moving their finger on the screen. This phase lasts until the user taps the ”Confirm” button to finished the initial phase.

2.2.2 Positioning Using SlidAR

SlidAR utilizes ray casting and epipolar geometry to adjust the position of virtual objects (Figs. 6 (a)-(c)). This process of SlidAR can be divided into 2 steps: (1) setting the 2D position and (2) adjusting the depth information. The 2D positioning process is performed during object initialization phase. When the user presses the desired location, the object appears correctly aligned with the target position from the current perspective. When the user presses the ”Confirm” button, the system casts a ray from the current camera pose to the created object to create the epipolar geometry.

When the user views the scene from a different viewpoint, the epipolar line is rendered as a 2D red line and is used to represent the depth information. The user can adjust the position of the annotation (depth information) by moving it along the epipolar line with a one-finger slide gesture. During the slide gesture, the user does not have to match the finger position with that of the virtual object.

However, if the 2D position is incorrect, the user can not use SlidAR to position the annotation to the desired location because the epipolar line does not intersect with it. In this case, the user can used the two-finger gesture to re-adjusting the 2D position by pressing on the desired location and then releasing the finger to recreate the epipolar geometry.

2.2.3 Orientation Control

The main idea behind SlidAR+ is to use the gravity information to assist in orientation control. I use the gravity information to assist in 2 processes: 1)

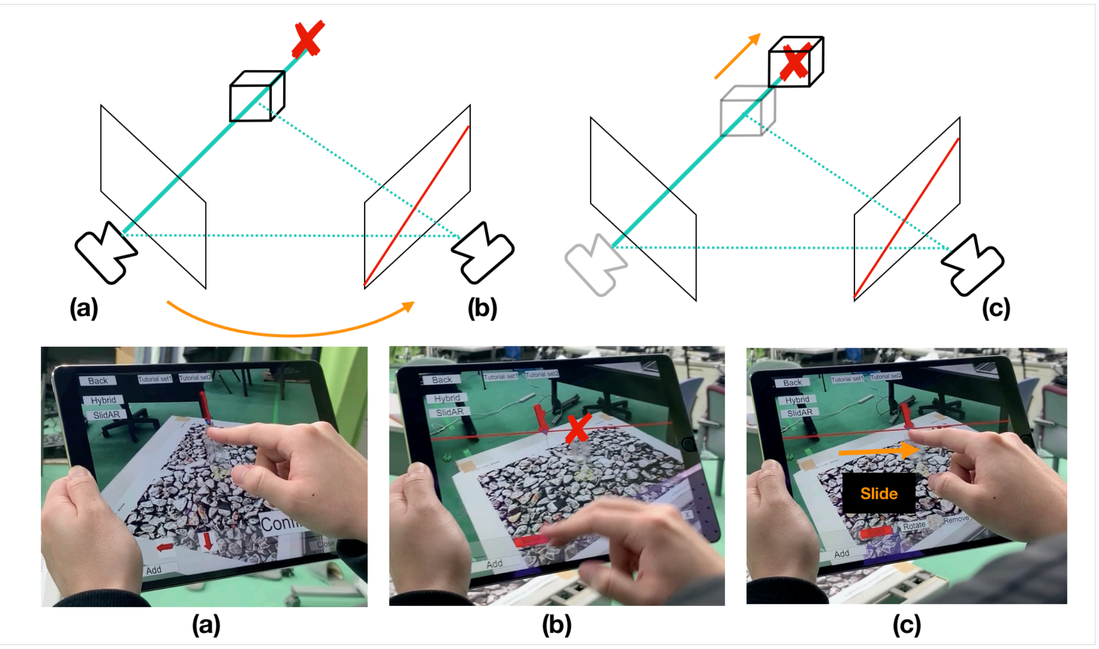


Figure 6: Positioning with SlidAR. (a) Upon creation of a virtual, the user inputs the 2D initial position. (b) After moving to another viewpoint, the position of the virtual object is misaligned because the depth information cannot be input during object creation. (c) The user corrects the position by sliding the object along the red epipolar line with a slide gesture.

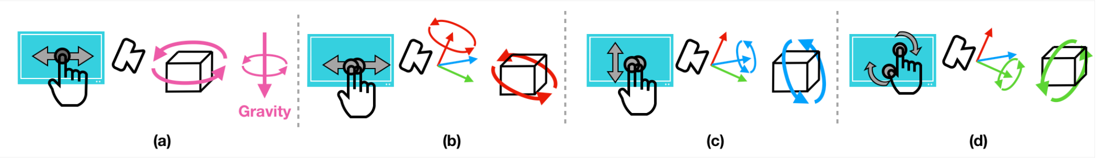


Figure 7: Orientation control in SlidAR+: (a) One-finger horizontal slide gesture to perform gravity constrained rotation. (b) Two-finger horizontal and (c) vertical slide gesture to rotate around the camera's x- and y- axes. (d) Two-finger twist gesture to rotate the object around the camera's z-axis

setting up the initial orientation , and 2) allowing the user to perform gravity constrained rotation.

During the initial placement process, the system automatically aligns virtual objects with the gravity vector based on user's selection, thereby effectively predetermining 2 DoFs. The last DoF is the rotation around the gravity vector. The user can then adjust the remaining DoF (rotation around the gravity vector) with a one-finger horizontal sliding gesture (Fig. 7 (a)).

Users can also manipulate all 3 rotational DoFs, if necessary, by using the two-finger vertical and horizontal sliding gestures for ARCBALL [92] rotation and a two-finger twist gesture [62] to rotate around the device's x-, y- and z-axis, respectively (Figs. 7 (b-d)).

2.3 Evaluation of SlidAR+

To evaluate the efficiency of SlidAR+ in 6 DoFs object manipulation tasks, I performed a user study to compare SlidAR+ with a state-of-the-art method, *Hybrid*. Marzo et al. [62] found that *Hybrid* is the most efficient object manipulation method compared to other device-centric and screen-based methods.

2.3.1 Methodology: *Hybrid*

Hybrid [62] combines device-centric movement and screen-based manipulation techniques. It uses the device-centric movement to control an object's position and screen-based gestures to control orientation. This allows the user to control all 6 DoFs at the same time without requiring the need to switch between the two methods.

We found Mazo et al.s' *Hybrid* suit our requirement very well as our task is to focus on manipulating 6 DoFs using tablet or 2D input devices .Hybrid efficiently controls the position of the object using the device movement which means users have to input only 3 DoFs for orientation through the 2D display. In their work, they compare *Hybrid* with device-centric movement and screen-based technique in 6 DoFs tasks. The results show that Hybrid performs the best among the three methods[62].

However, *Hybrid* did not include object placement and creation functionality.

Therefore, I have added this capability in my experiments. *Hybrid*'s workflow can also be divided into two 2 phases: 1) initial placement of the AR object, and 2) object manipulation.

Initial Placement of AR object

This is the same as in SlidAR+, as described in Section 2.2.2. The only difference is that instead of aligning the AR object to the gravity vector, *Hybrid* aligns it to the system coordinate.

Object manipulation: Positioning and Orientation

In *Hybrid*, a user can control both position and orientation at the same time without needing to switch modes, as in SlidAR+. To manipulate the object, the user first aligns the center of the screen with the intended object and taps with a finger anywhere on the screen. This fixes the object in the device's coordinate system. Now its position can be manipulated by moving the device (Fig. 8 (a)).

Orientation in *Hybrid* is controlled by combining ARCBALL and the two-finger twist gesture (Z-Rot). Both techniques rotate a virtual object relative to the camera axis. In ARCBALL, the user performs vertical and horizontal sliding gestures along the screen to rotate around the x- and y-axes while the two-finger twist gesture rotates the object around the z-axis (Fig. 8 (b)).

2.3.2 Experiments design

There are two main points I want to evaluate in this study: 1) the overall efficiency of SlidAR+ in 6 DoFs manipulation task, and 2) the affect on performance when the gravity control feature is added to SlidAR.

I therefore conducted two experiments: 1) a "positioning task" and 2) a "6 DoFs task". The first experiment will be used to confirm the performance of SlidAR as it has been implemented by us and evaluate it under condition that have not been covered in the previous study, as described in Section 2.3.2. The second experiment is the evaluates of SlidAR+ in a task that involves manipulating all 6 DoFs (positioning + orientation) tasks. By performing both the experiments,

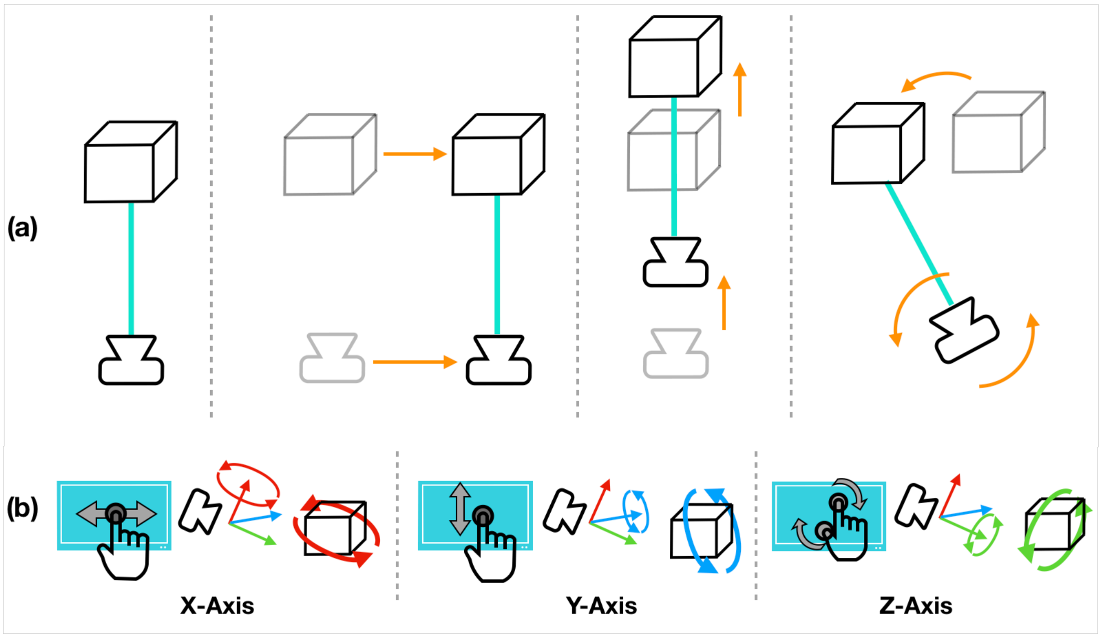


Figure 8: Object manipulation control in *Hybrid*: (a) Device centric movement position control (blue line indicates the fix distance between object and camera). (b) Orientation control in Hybrid: Horizontal and vertical slide gesture to rotate around the camera's x- and y- axes. Two-finger twist gesture to rotate the object around the camera's z-axis

I can compare the change in performance of SlidAR+ before and after adding orientation control to see the affect of my orientation control feature.

Depth cue using Shadow

One of the problems in object manipulating AR object with handheld devices is the lack of depth information as most of devices use a monoscopic display. To solve this problem, researchers have used the shadow cast by the virtual object onto the planar or ground surface to aid object placement.

When placing any AR content in the real-world scene, We can divided into two scenarios: 1) AR content is placed or attached on a real objects on the planar surface, or 2) AR content is placed on the mid-air or on a non-planar surface; for an example placing on an object protruding out of the wall. The difference between these two is the difficulty in using shadows to get depth cue information. In the first case, a user can easily get the depth information using the real-world object as a reference by directly matching the shadow with the base of the object on the planar surface. However, in the second case, the shadow might be projected on a difference surface, make it more difficult to to place the object correctly.

In the original SlidAR experiment [82], the effect of shadows could only be partially understood from the subjective user feedback. However, I would like to objectively confirm and verify the effect of shadows and depth cue on the performance of SlidAR+.

I therefore have performed the experiment under two conditions with different level of depth cue difficulties: 1) an ”easy condition”, and 2) a ”hard condition”. In the easy condition, the AR content is always placed on an object on the planar surface. In hard condition, the AR content is always placed mid-air or a non-planar surface.

Orientation related condition

In this part, I would like to explain the main factors that affect the orientation in my experiment. From my discussion in Section 2.1.2, my condition are related to the relationship between factor that affect the orientation and the gravity direction.

- Target pose is the position and orientation of the AR content that the user wants it to be. In this study I refer to it as the pose that the participants have to manipulate the AR content to match. Normally, target pose is aligned to a surface of the physical world environment. As described in Section 2.1.2, I assume that the target pose is aligned to the gravity vector in most cases. In summary, there are two conditions: 1) target pose is aligned (parallel or perpendicular) to the gravity, and 2) target pose is not aligned to the gravity.

- System coordinate refer to the coordinate created by a SLAM based AR application for the initial localization as discussed in Section 2, this coordinate is defined by detecting the physical world (mostly planar surface) and is used to determine the initial orientation of AR object. Normally, we would want the initial orientation to be as close to the target pose as possible or at least aligned to the same coordinate so as to reduce the number of angles or DoFs that may need to be adjusted. However, in some cases the system coordinate is not aligned properly, such as in case of an error during SLAM system initialization or if the planar surface is oblique to the alignment of target pose. This will increase the amount of time and effort needed to adjust the pose. In summary, there are 2 conditions: 1) the system coordinate is aligned to the gravity, and 2) system coordinate is not aligned to the gravity.

2.3.3 Overall Hypotheses

The hypotheses I evaluating were created based on the condition affecting the 6 DoFs, which can be divided into two groups: 1) position and 2) orientation.

- Translation-related condition
  In the easy condition, both SlidAR+ and *Hybrid* should have similar task completion times because with a depth cue *Hybrid* should be able to perform as fast as SlidAR+. In the hard condition, SlidAR+ should be able to complete the task faster than *Hybrid* because *Hybrid* requires the shadows for depth information. When shadows become difficult to observe, participants would have to spend more time adjusting depth and position.

However, this will not affect SlidAR+ as it does not rely on shadows to obtain the depth information (Table 1 (a)).

- Orientation related condition
  In case of both system coordinate and target pose are aligned to the gravity, SlidAR+ should show a faster completion time than *Hybrid* due to the gravity constrainted rotation feature of SlidAR+, which will allow users to complete the task faster than *Hybrid*'s ARCBall, even though both methods will have the same number of DoFs and angles to adjust. In the case where the target pose is aligned to gravity but system coordinate is not, I expect *Hybrid* to perform slower than system coordinate is aligned since the system coordinate will affect the initial pose in *Hybrid* and cause extra angle and DoFs needed to adjust. But this system coordinate will not affect SlidAR+ as it fix the initial pose to aligned to the gravity.

  In the condition where target pose not aligned to the gravity vector, both methods should show similar completion times as SlidAR+ has no direct advantage over *Hybrid* in such a case (Table 1 (b)).

From the above, I have 3 main hypotheses (Table 2):

- **H1**: SlidAR+ will perform better than *Hybrid* when target pose is aligned to the gravity vector.
- **H2**: SlidAR+ will perform better than *Hybrid* under the hard condition when the target pose is not aligned to the gravity vector.
- **H3**: SlidAR+ and *Hybrid* will have a similar performance (no difference) under easy condition when the target pose is not aligned to the gravity vector.

2.4 Experiment 1: Positioning Task

In Polvi et al.'s [82] experiment, the user had to place the AR content on a Lego structure. In this experiment, I wanted like to explore and evaluated the performance of SlidAR in the case where the mid-air object placement is needed.

Condition	
Easy	Similar performance
Hard	SlidAR+ is better

(a)

	Target pose aligned to gravity	Target pose NOT aligned to gravity
System coordinate aligned to gravity	SlidAR+ is better	Similar performance
System coordinate NOT aligned to gravity	SlidAR+ is better	Similar performance

(b)

Table 1: Summary of hypothesis: (a) Translation-related condition. (b) Orientation-related condition

Condition		Target pose aligned to gravity	Target pose not aligned to gravity
System coordinate aligned to gravity	Easy	SlidAR+ is better	Similar performance
	Hard	SlidAR+ is better	SlidAR+ is better
System coordinate not aligned to gravity	Easy	SlidAR+ is better	Similar performance
	Hard	SlidAR+ is better	SlidAR+ is better

Table 2: Overall hypothesis

I conducted this experiment following the same design as Polvi et al.'s experiment [82]. I measured efficiency on the basis of two aspects: 1) the average time taken to complete the task and 2) the average distance of device movement. I used a screen recorded to study the participants' behavior during the experiment.

2.4.1 Design

My experiment simulated a task support scenario, where participants were asked to place 3D annotations in the scene. I conducted the experiment in a laboratory environment with marker-based tracking for better control over all variables. This experiment used a within-subject design with two independent variables: 1) object manipulation methods, and 2) difficulty.

Independent Variables

- **Object manipulation method**
  I compared two object manipulation methods in this experiment: SlidAR and *Hybrid*. Both methods provide to create a 3D arrow annotations from the 3D assets provided by the system. All participants utilized both methods in a counterbalanced order.

- **Difficulty**
  This variable is used to described the difficulty in viewing and using shadow or depth cue while positioning the object. As described previously in Section 2.3.2, I defined two conditions based on difficulty: an "easy condition" and a "hard condition". In the easy condition, all of AR targets were placed on top of virtual pillars (height: 4 to 14 cm) that connect to the ground. Participants could adjust the position easily by matching the shadow of the AR object with the virtual pillar's base. Under hard condition, AR targets were placed on the top of floating pillars (height: 10 to 25 cm from ground). In this case, participants could not easily use the shadows to guess the position (Fig. 9).

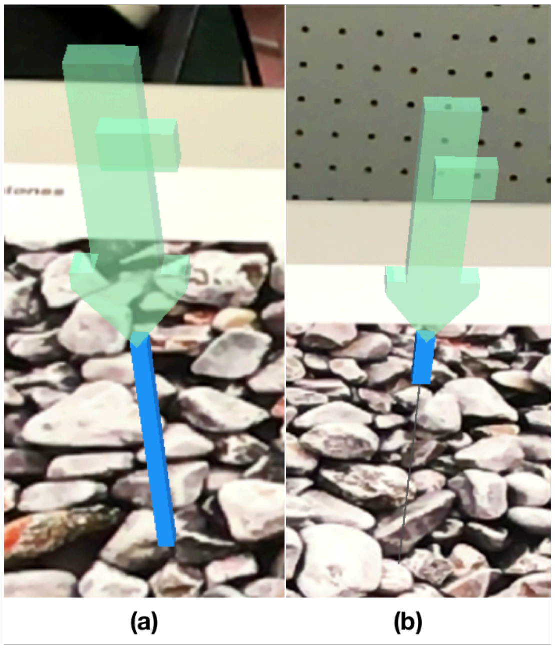


Figure 9: Picture of difficulty setup: (a) a long AR pillar which connect to the ground. (b) A floating AR pillar (Black line illustrate the high of pillar which is not show during experiment). Small green rectangle on arrow used to represent the direction of the arrow.

2.4.2 Experiment Platform and Setup

The handheld device used in this experiment was an Apple iPad Pro(2017) with a 1668 × 2224 pixels 10.5 inch display, Apple A10X CPU, and a weight of 477 g. The reason why we choose a tablet is because one of our goals is to apply this interface to an industrial AR application. In an industrial AR, the tablet is more preferable as it can provide more information on the larger screen at the same time than a smartphone. The system was usable only in portrait orientation with the back camera in the top-left corner. Furthermore, I used an AR marker for tracking the device pose and for defining the system coordinate in the AR application.

All tasks were conducted with the same setup, i.e., AR marker (80 × 60 cm) placed on a table (length = 80cm, width = 80cm, and height = 70cm). Participants were encouraged to walk and look around the table and tasks area from different angles and viewpoints. This setup were used in both the experiments.

2.4.3 Hypothesis

I have three hypotheses for this first experiment based on the ”Translation-related condition” in Section 2.3.3.

- **E1-H1**: SlidAR and *Hybrid* should have similar completion times under easy condition.
- **E1-H2**: SlidAR will be faster than *Hybrid* under the hard condition.
- **E1-H3**: SlidAR will require less device movement than *Hybrid*.

Hypotheses **E1-H1** and **E1-H2** are based on the hypotheses shown in table 1 (a). In **E1-H3**, I expected SlidAR to utilize smaller device movements to accomplish the tasks as participants using *Hybrid* would have to move the device in order to positioning the object, whereas SlidAR participants only need to move the device once to the change viewpoint before they begin positioning.

2.4.4 Tasks

The participants were asked to create and place an AR arrow (represented as a red AR arrow similar to the one shown in Figure 9 (b)) and move it to the correct position. Out of all the virtual objects, only the created AR arrow casted a computer-generated shadow. Participants can receive a depth cue by trying to match the shadow from the created AR arrow with the base of the virtual pillar. Each participant completed two tasks per method (four tasks per participant in total). Each task consisted of five trials or five target AR arrows that were highlighted as a translucent green AR arrow presented one at a time. To completed each trial, participants had to create an AR arrow and align its position with the shown target. The system automatically checked the alignment of the user created arrow with target AR arrow by comparing their positions. The task was completed if the difference between the current object position and target position was within a set margin (2 cm) for 1 second. When one trial was completed, both arrows disappeared and the next trial was started. Participants received a notification once they completed all five trials. In all, participants had to align 20 target annotations.

2.4.5 Procedure

The experiment took approximately 40 to 60 minutes to complete per participant. First, each participant was tutored for up to 10 minutes (explanation and practice level) on the first method (depending on the order of each participants). I instructed participants on the following steps: (1) how to create an arrow, (2) how to adjust and correct the position, and (3) a way to use each method effectively. Participants are free to grasp and hold the device as they are favored and comforted.

Next, each participant spent approximately 5 to 10 minutes (depending on individual skill) to completed all tasks using the first method (approximately 2 to 5 minutes per task). After the experiment, the participant was asked about their opinion of the method, and their answeres were recorded on a HARUS (Handheld Augmented Reality Usability Scale) questionnaires [90] (Table 3), that recorded subjective feedback in two aspect: manipulability and comprehensibility, and free-form written comments. This process took approximately 5 minutes, followed by a small break. Then, the tutorial for the second method started with the same procedure as the first.

All measurement data were captured automatically by the system. And I

Manipulability:
Q1. I think that interacting with this application requires a lot of body muscle effort.
Q2. I felt that using the application was comfortable for my arms and hands.
Q3. I found the device difficult to hold while operating the application.
Q4. I found it easy to input information through the application.
Q5. I felt that my arm or hand became tired after using the application.
Q6. I think the application is easy to control.
Q7. I felt that I was losing grip and dropping the device at some point.
Q8. I think the operation of this application is simple and uncomplicated.

Comprehensibility:
Q9. I think that interacting with this application requires a lot of mental effort.
Q10. I thought the amount of information displayed on screen was appropriate.
Q11. I thought that the information displayed on screen was difficult to read.
Q12. I felt that the information display was responding fast enough.
Q13. I thought that the information displayed on screen was confusing.
Q14. I thought the words and symbols on screen were easy to read.
Q15. I felt that the display was flickering too much.
Q16. I thought that the information displayed on screen was consistent.

Table 3: HARUS Questions

screen-recorded all of the operation data for each trial. As for the time data, I divided it into two parts: (1) Authoring time: the time participant spent in adjusting the initial 2D position. This was recorded after participant created an AR arrow until the end of initial phase. (2) Editing time: the time participant spent adjusting an AR arrow to the correct position. This was automatically recorded at the time between the end of initial phase until the trial was completed. I also measured the device's movements based on the relative position of the device's camera to the marker. Every 30 frames, the trajectory between the current and previous poses was added to the total device movement.

2.4.6 Participants

I recruited a total of 12 participants from the local university (8 males and 4 females; average age: 24 years (SD = 1.4); range: 23-27 years). I asked the participants about their experiences with AR applications and 8 participants reported having using an AR application previously whereas 4 had never used any AR application before. I also asked the participants about their pre-existing knowledges in 3D manipulation: 5 participants were familiar, 2 had moderate knowledge, and 5 had no experience.

2.4.7 Results

For my analysis, I first ran a normality test on the data. The results from Shapiro-Wilk test showed that the data violated normality ($p < 0.05$). So, I used non-parametric Wilcoxon test for the analysis of the data.

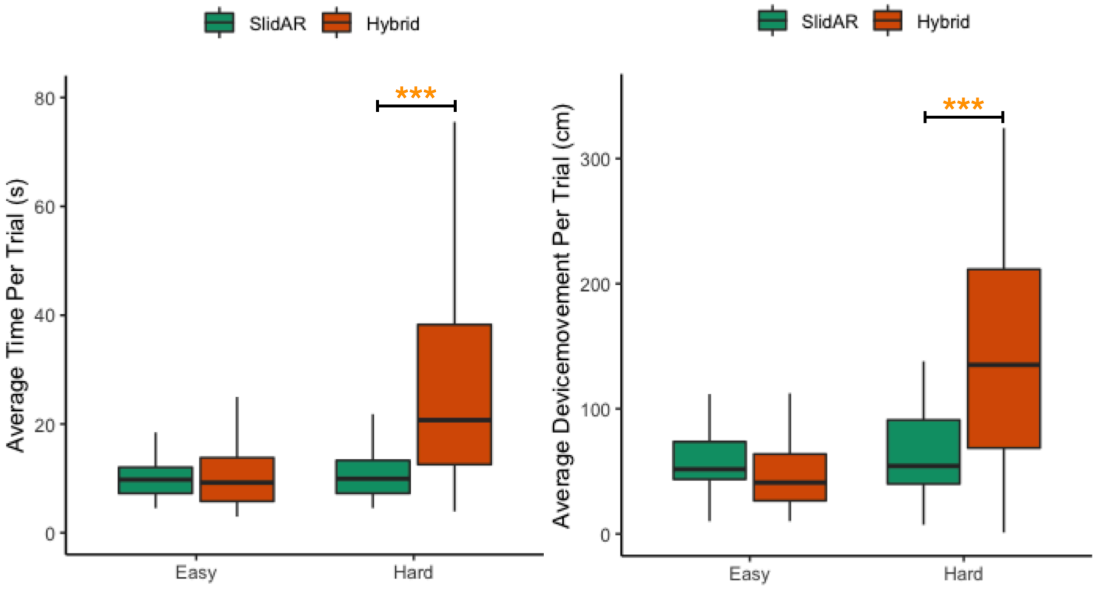
I noticed that SlidAR (Mdn = 9.94) completed tasks significantly faster than *Hybrid* (Mdn = 21.71) under the hard condition, $z = 5.072, p < 0.001, r = 0.655$. However, I found no significant difference between SlidAR (Mdn = 9.76) and *Hybrid* (Mdn = 9.23) under the easy condition, $z = 0.611, p = 0.54, r = 0.078$ (Fig. 10a). Next, I investigated the experiment results under the easy condition and found that SlidAR (Mdn = 2.92) required significantly more time for 2D positioning than *Hybrid* (Mdn = 0.1), $z = 6.14, p < 0.001, r = 0.793$. However, SlidAR (Mdn = 6.22) showed a significantly less time in editing mode than *Hybrid* (Mdn = 8.91), $z = 2.88, p = 0.003, r = 0.372$.

For the device movement, I can not find any significant difference between SlidAR+ (Mdn = 51.72) and *Hybrid* (Mdn = 41.035) in easy condition, $z = 1.899, p < 0.057, r = 0.245$. However, SlidAR (Mdn = 54.31) recorded significantly smaller device movements than *Hybrid* (Mdn = 179.92), under the hard condition, $z = 5.33, p < 0.001, r = 0.688$) (Fig. 10b).

Upon analyzing the subjective feedback, I could not find any significant difference in the total score, manipulability, or comprehensibility between SlidAR and *Hybrid* (Fig. 11). In the free-form written feedback, 7 participants preferred SlidAR and 5 preferred *Hybrid*.

2.4.8 Discussion

Under the easy condition, there was no significant difference in completion time between SlidAR and *Hybrid*, which support the **E1-H1**. This happened because most participants spent a lot of time in positioning the arrow in SlidAR whereas



(a) Average completion time per trial.

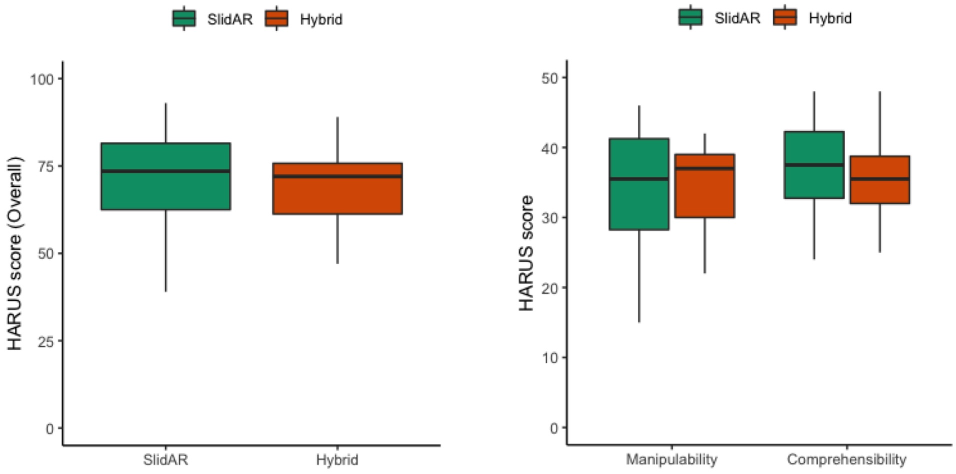
(b) Average device movement per trial.

Figure 10: Result of objective measurements: (a) task completion time and (b) device movement. Connected bar represents significant difference (* = significant at 0.05 level, *** = significant at 0.001 level).

in *Hybrid* they could just tap on screen instantly and adjust the position after that. However, SlidAR performed better in the editing mode(3D position) as SlidAR can correct an object's position just via a single slide gesture whereas *Hybrid* requires adjustment of all 3 DoFs.

Results under the hard condition were clearer and I could see that SlidAR performed better and required less device movements which supports **E1-H2**. A plausible reason of this is that SlidAR does not require any additional depth information whereas participants take a longer time to move and check the current position or guess by looking at the shadows while using *Hybrid*.

The results show that SlidAR+ was significantly faster than *Hybrid* only under the hard condition and not in easy condition, which negates the **E1-H3**. One of the reason for this is that under the easy condition, participants can match the shadow with the base of pillars in order to get depth information instead of moving to other viewpoints when using *Hybrid*. Whereas under the hard



(a) HARUS total score.

(b) HARUS individual score.

Figure 11: Result from HARUS questionnaire: (a) total score and (b) manipulability and comprehensibility score.

condition, participants need to move to other viewpoints to get depth information and this process may need to repeated many times during one trial. However, in SlidAR participants have to move only once.

Upon analyzing the subjective feedback, I did not find any significant differences between SlidAR and *Hybrid* in term of any category of HARUS questionnaire, which was different from the results of the Polvi et al.'s study [82]. We believe this might happen due to the duration of task, weight and size of device causing a negative effect on SlidAR more than in Hybrid.There were 2 main reasons why 5 participants of the 12 participants still preferred *Hybrid* over SlidAR. First, even through many participants comment that SlidAR is easier to use, *Hybrid* offered more freedom of control and they felt more engaged and entertained when using it. The second reason was related to the difficulty and fatigue induced when holding the device. From observation all participants in each method have the following manner. During the initial placement process, both methods utilized a single hand holding gesture where participants had to use one hand to hold the device and the index finger of the other hand to interact with the device. Participants spent less time in this process in *Hybrid* as it does not require an accurate 2D position as SlidAR. For the process of moving and changing the viewpoint to observe and check the correctness, participants were likely to hold the device with two hand gestures as it was more comfortable in both methods. For the position adjustment, SlidAR used the single hand gesture with the other hand as support to position the AR content. While in *Hybrid*, participants could operate by holding the device with two hands while using their thumbs to interact with the screen. As the single hand holding causes more physical fatigue on big and larger devices than with two hands holding, resulting in more physical fatigue in SlidAR than *Hybrid*. However, many participants felt that the tasks were too short to feel any difference in terms of fatigue between both methods, but this might change if the task become longer. I also received comments about using smaller devices such as mobile phone, rather than a tablet.

Overall, I found that the results supported **E1-H1** and **E1-H2** but not **E1-H3**.

2.5 Experiment 2: 6DoFs Task

From the previous experiment, I found that SlidAR is significantly faster than *Hybrid* under the hard condition but both have similar performance under the easy condition. However, the task in this experiment, *Hybrid* has an advantage in that it allows control over both position and rotation at the same time, while SlidAR+ controls them separately.

2.5.1 Design

The basic design of this experiment is similar to the previous experiment. Participants had to create an AR arrow, place it into the scene, and adjusting its pose to match a target (position and orientation). I used the same platform and equipment setup as in the previous experiment, except with an additional variable that effects orientation. In this experiment I used a within-subject design with three independent variables: object manipulation methods, target pose, and coordinate system as I described in Section 2.3.2. For evaluation, I measure same efficiency in terms of time, device movement, and subjective feedback just as in the previous experiment.

Independent Variables

- **Target pose**
  This variable describes the alignment of the target pose relative to the direction of gravity: 1) target pose is parallel or perpendicular to the direction of gravity vector or 2) target pose is not parallel or perpendicular to the direction of gravity vector.

- **The system coordinate**
  This variable is the relationship between the the system coordinate of the AR application and the gravity direction as I described in Section 2.3.2. There are two possibilities for this variable are as follows: 1) The system coordinate of the AR application is aligned to the gravity vector, or 2) The system coordinate of the AR application is not aligned to the gravity vector.

The dependent variables are objective results consisting of task completion time (seconds) and device movement distance (cm). I also collected subjective feedback following the experiment using HARUS [90] and free-form written feedback.

Experimental Conditions

In the experiment, I have four conditions (target pose × coordinate system) for each manipulation method.

Condition 1: The system coordinate is aligned to the gravity vector and the target annotation is either parallel or perpendicular to gravity.

Condition 2: The system coordinate is aligned to the gravity vector and the target pose is not parallel or perpendicular to gravity.

Condition 3: The system coordinate is not aligned to the gravity vector and the target pose is either parallel or perpendicular to gravity.

Condition 4: The system coordinate is not aligned to the gravity vector and the target pose is not parallel or perpendicular to gravity.

2.5.2 Hypothesis

On the basis of the design of SlidAR+ and *Hybrid*, I had three hypotheses for this experiment as discussed in Section 2.3.3:

- **H1**: SlidAR+ will perform better than *Hybrid* when target pose aligns to the gravity vector.
- **H2**: SlidAR+ will perform better than *Hybrid* when the target pose does not align to the gravity vector under the hard condition.
- **H3**: Both SlidAR+ and *Hybrod* will perform similarly (no difference) when the target pose is not aligned to the gravity vector under the easy condition.

2.5.3 Tasks

As previously mentioned, the task involved in this experiment were similar to experiment 1, which include placing and adjusting an AR arrow to the target pose (position and rotation). Each participant had to completed four tasks per method (total eight tasks per participant). And each task consisted of six trials or six target arrows, with the first three target arrows were set on long pillars (”easy trial”) and the last three targets were set on floating pillars (”hard trial”), as in the previous experiment. In order to complete the task, participants had to complete all trials by aligning the created arrow with the target one. The system determined whether both the user created and target arrows are aligned or not. The setting margin for the correctness of pose in this experiment were set at 2 cm for translation and 12° for orientation. In all, participants had to align 48 target arrows.

2.5.4 Procedure

I divided the experiment into two sections, one per each manipulation method. Each section took approximately 50 to 60 minutes per participant. First, a participant spent time up to 20 minutes being tutored (explanation, presentation, and practice level) on the first method (depend on the order of each participants). I instructed participants on the following steps: (1) how to create an arrow, (2) how to adjust, and correct the position, and (3) a way to use each method effectively. Again, we did not specify the way to grasp and hold the device, participants can operate as they wish.

Next, the participant spent approximately 20 to 50 minutes (depending on individual skill) completing all tasks using the first method (approximately 5 to 10 minutes per task) followed by a 5 minutes break between the tasks. After the experiment, the participants were asked about their opinion of the method; they also answered an questionnaires that recorded their subjective feedback and free-form written comments. Next, the participant took a small break. This process took approximately 15 to 20 minutes and then the whole process was repeated for second section.

All data were measure automatically by the system, and I screen-recorded all of the operation data for each trial. I divided time data into 2 parts: (1) Authoring time: time spent by participant in adjusting the 2D initial position. This was recorded after a participant created an AR arrow until the end of initial phase. (2) Editing time: time spent by a participant adjusting an AR arrow to the correct position. This was automatically recorded as the time between the end of initial phase until the trial was completed. I also measured the device movement based on the position of the device's camera relative to the marker. Every 30 frames, the trajectory between current and previous pose was added into total device movement. Finally, I collected subjective feedback after the experiment using HARUS [90] and free-form written feedback.

2.5.5 Participants

I recruited a total of 16 participants for this experiment (11 males and 5 females; average age: 24 years (SD = 1.4); range: 23-27 years) I asked the participants about their experience with AR application and 13 participants reported having used an AR application previously but 3 had never used an AR application before. I also asked the participants about any pre-existing knowledge in 3D manipulation: 10 participants were familiar, 1 had moderate knowledge, and 5 had no experience. 10 participants out of the 16 participants had participated in the previous experiment as well, however, there was at least a 1 day break between each experiment.

2.5.6 Results

Shapiro-Wilk test showed that the data in this experiment 2 also violated normality ($p < 0.05$). Therefore, I used non-parametric Wilcoxon test to analyze the data. I consider results significant for $p < 0.05$.

Task efficiency

In the overall completion time (easy + hard trials), I found the significant differences in Condition 1 and 3 wherein SlidAR+ completed the tasks faster than *Hybrid* (Con1: Mdn = SlidAR+ **(S)** 16.63(s) vs *Hybrid* **(H)** 44.45,$z = 7.32, p < 0.001, r = 0.74$; Con3: Mdn = **(S)** 16.84 vs **(H)**35.29,$z = 6.05, p < 0.001, r = 0.61$).

For a detailed analysis of the average completion time data, I focused only on the performance between SlidAR+ and *Hybrid* in the same condition with the same trials settings, and I did not compared the data between difference condition or trials. We do not report the results of authoring and editing time as the time spent in authoring mode is very small and similar to experiment 1. The results of editing time are also similar to the overall task completion time.

In easy trials, SlidAR+ performed significantly faster than *Hybrid* when the targets were aligned to the gravity vector (Con1: Mdn = **(S)** 16.1 vs **(H)** 32.52, $z = 7.32, p < 0.001, r = 0.74$; Con3: Mdn = **(S)** 17.04 vs **(H)** 25.81, $z = 6.05, p = 0.009, r = 0.61$). However, in Conditions 2 and 4, SlidAR+'s performance was significantly slower than *Hybrid* (Con2: Mdn = **(S)** 42.89 vs **(H)** 31.63, $z = 2.3, p = 0.02, r = 0.03$; Con4: Mdn = **(S)** 47.33 vs **(H)** 28.84 $z = 3.81, p < 0.001, r = 0.015$). As for the hard trials, SlidAR+ completed tasks significantly faster than *Hybrid* in the Conditions 1, 3, and 4 (Con1: Mdn = **(S)** 17.65 vs **(H)** 57.52, $z = 5.52, p < 0.001, r = 0.79$; Con3: Mdn = **(S)** 16.45 vs **(H)** 53.19, $z = 5.48, p < 0.001, r = 0.79$, Con4: Mdn = **(S)** 3.45 vs **(H)** 51.57, $z = 2.64, p = 0.008, r = 0.38$). However, I could not find any significant difference between SlidAR+ and *Hybrid* under Condition 2 (Mdn = **(S)** 42.98 vs **(H)** 53.19, $z = 1.6, p = 0.109, r = 0.23$) (Fig. 12).

Analyzing the device movement results, I found that SlidAR+ required significantly less movement than *Hybrid* in the Conditions 1,3 and 4 (Con1: Mdn = **(S)** 89.08(cm) vs **(H)** 303.39, $z = 7.21, p < 0.001, r = 0.73$; Con3: Mdn = **(S)**

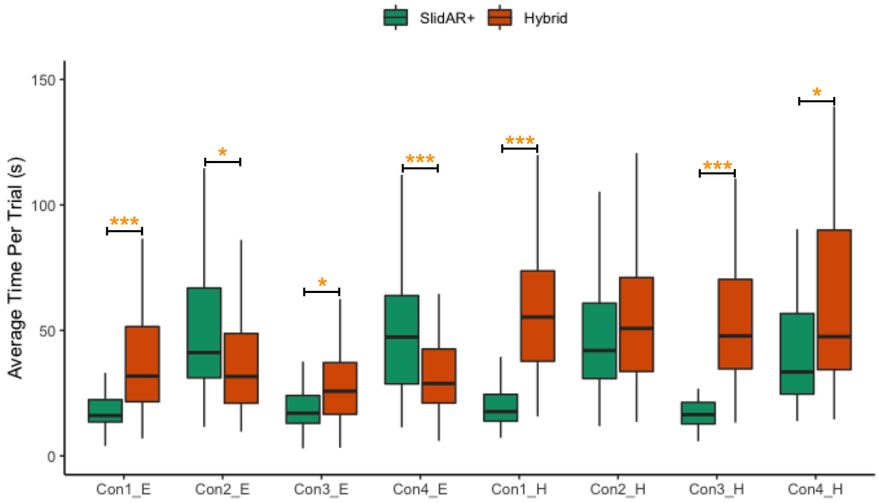


Figure 12: Result of objective measurements, an average task completion time . Connected bar represents significant difference (* = significant at 0.05 level, *** = significant at 0.001 level).

81.1 vs **(H)** 239.8, $z = 6.57, p < 0.001, r = 0.67$; Con4: Mdn = **(S)** 209.08 vs **(H)** 251.25, $z = 2.31, p = 0.02, r = 0.23$). But this was not the case with Condition 2 (Mdn = **(S)** 265.91 vs **(H)** 231, $z = 0.22, p = 0.8, r = 0.022$).

When analyzed the data in more detail, I found that under the easy trials, there was a significant difference between SlidAR+ and *Hybrid* wherein SlidAR+ required smaller device movement than *Hybrid* under Conditions 1 and 3 (Con1: Mdn = **(S)** 81.29 vs **(H)** 189.12, $z = 4.27, p < 0.001, r = 0.61$; Con3: Mdn = **(S)** 85.93 vs **(H)** 132.57, $z = 2.08, p = 0.03, r = 0.3$) but not under Condition 2 and 4 (Con2: Mdn = **(S)** 244.81 vs **(H)** 173.36, $z = 1.92, p = 0.053, r = 0.27$; Con4: Mdn = **(S)** 213.31 vs **(H)** 186.84, $z = 1.63, p = 0.1, r = 0.23$). Under the hard trials, SlidAR+ required significantly smaller device movement than *Hybrid* in every condition (Con1: Mdn = **(S)** 97.97 vs **(H)** 444.39, $z = 5.65, p < 0.001, r = 0.81$; Con2: Mdn = **(S)** 272.34 vs **(H)** 378.53, $z = 2.03, p = 0.04, r = 0.29$; Con3: Mdn = **(S)** 80.43 vs **(H)** 372.97, $z = 5.93, p < 0.001, r = 0.85$, Con4: Mdn = **(S)** 202.7 vs **(H)** 415.42, $z = 3.89, p < 0.001, r = 0.56$) (Fig. 13).

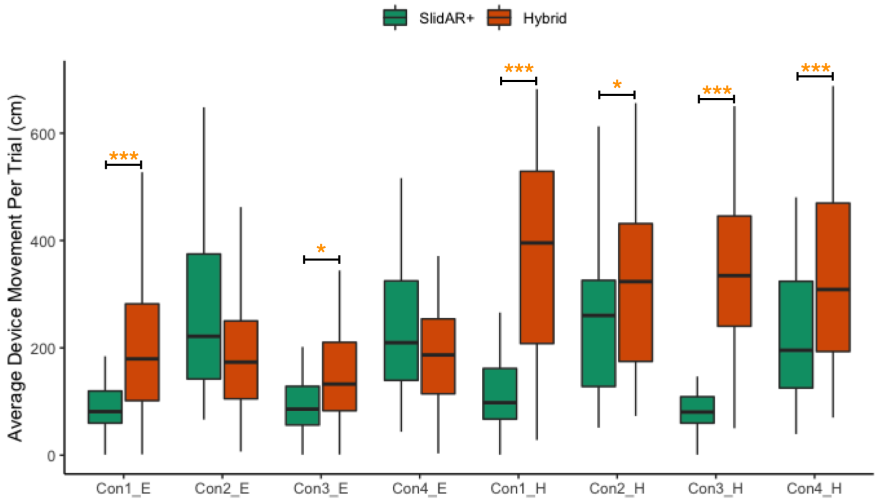
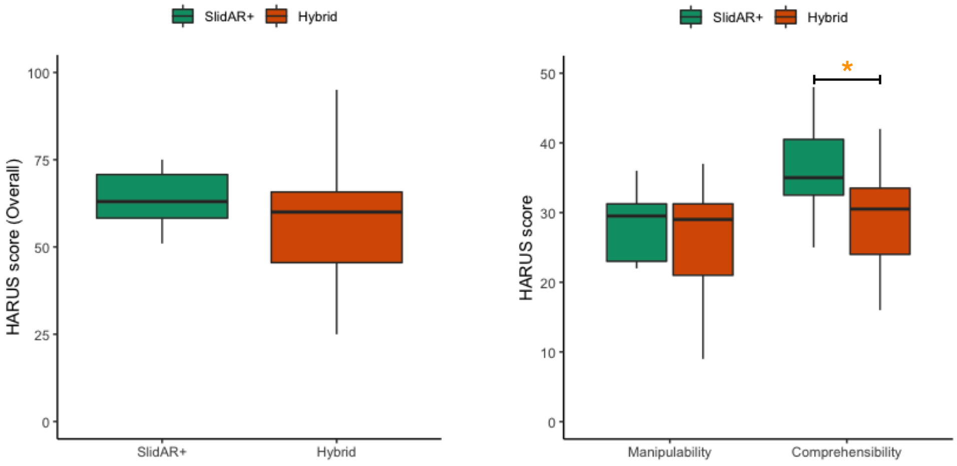


Figure 13: Result of objective measurements, an average Device Movement . Connected bar represents significant difference (* = significant at 0.05 level, *** = significant at 0.001 level).



(a) HARUS total score.

(b) HARUS individual score.

Figure 14: Result of HARUS questionnaire: (a) total score and (b) manipulability and comprehensibility score. Connected bar represents significant difference (* = significant at 0.05 level, *** = significant at 0.001 level).

Subjective feedback

For subjective feedback, I measured user preference through the HARUS scale. I ran paired Wilcoxon signed rank test to analyze the data. Analyzing the results, I could not find any significant difference between score of SlidAR+ and *Hybrid* on the overall (Fig. 14a) and manipulability. However, SlidAR+ (Mdn = 34) was scored significantly higher than *Hybrid* (Mdn = 31) in comprehensibility ($z = 2.833, p = 0.004, r = 0.731$) (Fig. 14b). An in-depth analysis of each questions revealed a significant difference with $p < 0.05$ on Q4, Q9, Q11, and Q13.

In the free-form written feedback, 13 participants preferred SlidAR+ whereas only 3 preferred *Hybrid*. Many comments were similar to the previous experiment, wherein participants felt that SlidAR+ was easier to use and understand. They could easily correct the position using fewer inputs by using SlidAR, and they did not have to move a lot. Many participants reported that they felt more engaged, entertained and had more freedom to control while using *Hybrid*. They also reported about the difficulty in holding the device while carrying out the tasks using SlidAR+, something that they did not experience with *Hybrid*.

2.5.7 Discussion

The results of the first experiment showed SlidAR+ having better performance than *Hybrid* under the hard condition but not under the easy condition. By including the orientation control in the tasks in the second experiment, I could see a change in terms of performance between SlidAR+ and *Hybrid* based on the conditions.

In overall performance (easy and hard trial), SlidAR+ showed a better completion time than *Hybrid* under Conditions 1 and 3, in which the targets were aligned to the gravity vector. In these conditions, the participants could easily correct the AR object's orientation using gravity constrainted rotation. Whereas in *Hybrid*, the participants could only perform rotation based on the device's camera perspective. However, I could not find any significant differences in completion time between the two methods when the targets were not aligned to the gravity vector (Condition 2 and 4), which support **H1**.

Under the easy trials, I observed a change in performance when compared to the first experiment, where in SlidAR+ now required significantly less time to completed the tasks than *Hybrid* under Conditions 1 and 3. However, *Hybrid* performed significantly faster than SlidAR+ under Conditions 2 and 4. In these conditions, SlidAR+ had no advantage over *Hybrid* in terms of rotaion control using gravity; thus, both methods had to manipulate the same amount of DoFs. One of the plausible reason is the separation of control scheme between the two methods. In *Hybrid*, the participant can control both position and orientation at the same time without need to switching the control mode, unlike in SlidAR+. Hence, **H3** is rejected.

In the trials under the hard condition, I found a significant performance difference with SlidAR+ performing faster than *Hybrid* in Condition 1, 3, and 4 but not in Condition 2. Hence, **H2** is rejected. We believe that the main reason SlidAR+ performed better in most of hard trials is because of the positioning, as I found in the first experiment. However, in the Condition 2 I could not find any significant difference between both methods. One of the plausible reason for this is in condition 2 the system coordinate was aligned to the gravity direction for both methods. So, the participant knew the initial orientation of created object and was able to predict the next step that need to be performed. Unlike in the Condition 4, the initial pose of the created targets in *Hybrid* appeared random or unpredictable for the participants, and this might have caused the performance of *Hybrid* to worsen under Condition 4 in comparison to Condition 3. However, I could not find any significantly evidence to support this and further investigation is required. Overall, the completion time results support only **H1**.

As for the device movement, SlidAR+ required significantly less movement in overall data (easy and hard trials) under Conditions 1,3 and 4. I also found the change in the results of easy trials under Conditions 1, 3, and 4. I also found the change in the results of the easy trials comparison to the first experiment, wherein I now found that SlidAR+ required significantly less device movement than *Hybrid* under Conditions 1 and 3 (where the target pose were aligned to the gravity vector). For SlidAR+, the participants needed to change their viewpoint only once to adjust the position. In the case of targets aligned to gravity vector, the participants can adjust the orientation without needing to move the camera/device to change the viewpoint. Under the hard trials, SlidAR+ required significantly less movement in every Condition which is similar to the results of the first experiment. This shows that the positioning process has a larger influence over device movement than orientation under hard trials.

The subjective results from HARUS showed that SlidAR+ is required significantly less mental effort and was easier to use than *Hybrid*, which is reflected by the comprehensibility score. But there was no significant difference between the two methods in terms of manipulability. I believe that this was mainly because the ways participants holding the device in each method have balanced out the physical effort to complete the task. For the translation, the holding behavior of participants is similar to the experiment 1 for both SlidAR+ and *Hybrid*. In SlidAR+ participants performed the task using one hand holding gesture for the whole process of rotation. However, in Hybrid participants can perform rotation using one or two hand holding gestures based on their preference. SlidAR+ might require less physical effort to input and operate but the way of holding in SlidAR+ might require more physical effort than one in *Hybrid*. One hand holds a big device such as an IPad Pro requiring more physical effort and less comfortable than two hand holding. Even though it is not significant, SlidAR+ likely has a higher average score in questions related to fatigue on arms and hands such as Q5 (SlidAR+: Avg 4.68, std 0.41; Hybrid: Avg 3.93, std 0.42). This is the reason we believe why the result of manipulability has no significant difference between two methods. Further analysis of HARUS scores revealed that participants found SlidAR+ to be simpler and easier to used as I observed a significant difference in Q4. SlidAR+ scored less in negative questions in Q9, Q11, and Q13 (Table 3), revealing that participants SlidAR+ required less mental effort than *Hybrid* to accomplish the same task.

To summarize, in the 6 DoFs tasks, SlidAR+ showed a faster completion time than *Hybrid* when the targets were aligned to the gravity vector. However, if they were not, SlidAR+ showed similar to *Hybrid*, but under hard trials only. It was slower in the easy trials. By combining the results of both the experiments, I can conclude that the gravity constrained orientation control feature can improve the performance of SlidAR when the targets are aligned to the gravity vector. However, it also worsens performance if the targets are not aligned.

I believe that SlidAR+ can be useful in the situations where it is hard to observe the depth cue/information. It can also be useful in situations where users have movement limitations in terms of size and control of the actual annotation space, as with remote collaboration scenarios. The subjective feedback and comments from the participants also suggest that SlidAR+ can help improve usability and reduce the mental effort required in placing 3D annotations.

Limitations

SlidAR+ was designed for users who want to place 3D content that is aligned either parallel or perpendicular to the direction of gravity. However, in cases where the target pose is not aligned to the direction of gravity, SlidAR+ still requires the user to control all 3 DoFs in orientation. SlidAR+ also requires the user to manually switch between position and rotation modes.

In the experiments, as I used a translucent object as the target, some of the participants found it difficult to see its orientation and this might have affected the results. Also, there were no real-world physical objects other than the pattern of the marker on the table that can be used as reference. The tracking quality and the error during the experiment might also effect the performance. Hence, the performance of both the methods may differ in other environment setups, the scale of the environment, and the number of object to manipulate at time same time might affect the results. Additionally, We do not record the amount of touch interaction in our experiment as we focus on the task completion time and user mental load rather than the number of interactions. However, SlidAR+ requires touch input to control both position and rotation. This makes SlidAR+ require more multiple combinations of touches compared to other methods that utilizes device-centric movement. This might cause SlidAR+ to require more learning time compared to methods with fewer touch inputs. Finally, the number of participants in my experiments are only 10 for experiment 1 and 16 for experiment 2.

The current SlidAR+ UI is not conducive for large devices because both of the participant' hands are needed to hold the device, which make it inconvenient to use SlidAR+. I believe that if I use a lighter and smaller device, the subjective feedback results for SlidAR+ will be improve.

2.6 Future Work

In future work, I advise improve the SlidAR+ UI in order to better support devices with large screens. And try to find other techniques to rotate virtual object (when the target pose is not parallel or perpendicular to the direction of gravity). I also think it might be better to explore SlidAR+ on other devices such as head-mounted displays.

2.7 Chapter Summary

In this Chapter, I presented SlidAR+ an object manipulation method for HAR applications. SlidAR+ utilizes ray-casting and epipolar geometry for positioning and gravity constrained orientation adjustment of virtual objects. SlidAR+ has been designed to minimize the number of inputs necessary to adjust the pose of the virtual object. My experiments showed that SlidAR+ is more efficient than a state-of-the-art object manipulation method. It showed faster completion times, required smaller device movement when AR contents were to be placed aligned to the ground, and exhibited significantly better comprehensibility. I expect SlidAR+ to be used as an alternative choice for many AR object placement scenarios such as task support, navigation design, or virtual object manipulation for entertainment.

3. Investigate the effect of camera latency on 2D display for micro-task

With the increase of complex machines that require expert knowledge or high skill for operation and maintenance in our daily life, HAR for task support becomes a popular method due to the availability and widespread of the handheld device. Although HAR have proved to be an effective method [32, 35], the majority of handheld device utilizes a see-through display which requires users to observe the scene from the perspective of a camera lens. This limitation can affect users' performance due to distortions in a visual representation such as [19, 20, 80] (i) distortion of binocular disparity,(ii) introduction of geometric distortions, and (iii) system latency. The first two distortions mainly cause by depth perception and dual-view problems from a 2D display such as a mobile phone. Many research has addressed [7, 10, 21, 24, 35] and proposes many solutions to overcome this problem [19, 20]. However, several aspects of latency remain unexplored.

Many factors can cause latency in HAR task support systems such as network latency, video processing, the rendering process, or the process of AR registration. The commercial handheld devices can have up to 125 ms delay for touch screen input-output latency [46]. Even with the medical-grade equipment for robotic and laparoscopic surgery can have camera latency up to 90 ms [53, 71]. It would not be surprising if the commercial-grade handheld device with AR applications will have higher system latency.

There are several situations where camera latency affects the task using a handheld device. I divided it into two groups based on the mobility of the handheld device and target object;

- Static situation
  Both the device and the target object are unmoveable in this situation. For example, the mobile phone set up on a static holder while performing a task on a rigid object. In this situation, there is a gap or delay between the movement of the user's hand on the physical world and the display screen caused by the camera latency.

- Dynamic situation

> In a dynamic situation, either the device or the target object is moveable. For example, the user holds the device while performing the task or the target object is a deforming or moving object such as a human body part. Apart from the gap between the hand movement, there are furthermore several gaps or delays that happen because of the camera latency. One of these is the registration of AR contents when the target object is deforming or moving causes the AR contents lagging behind the intended position. Other is the gap or difference between the shape of the target object present in the physical world and the display screen.

Many studies report that latency has an impact on the users' perception of responsiveness [69], increase task duration, and reduce accuracy [53]. Many solutions to reduce the effect have been proposed [104]. However, most of the work in this area has focused on the touch screen input latency in the handheld device. Or camera latency (video lag) with an indirect input device such as a robot arm in the laparoscopic surgery system. The questions of the effect of camera latency while performing the task looking through the camera display in the handheld device where users use their hand to perform the task behind the camera. Thus, I would like to address these following research questions:

- What is the minimum latency user can perceive?
- How does latency affect time and accuracy?

In this study, I would like to answer these questions by investigating the effect of camera latency on users following AR instructions using hand to performing the task behind the camera in handheld devices. For this case study, I choose a static situation with a micro-task scenario such as circuit board debugging or soldering as a target for video-see though AR task support. I choose the static situation in this case study because there is only a single factor affected by the latency before continuing to study in a dynamic situation in future work. The reason that I choose the Micro-task because the previous studies show that latency affects the performance in surgery tasks[53, 71] which is a type of task that requires high accuracy and precision. I believe that this type of task will be easier to see the effect of latency on user performance. So, I decided to used Micro-task as a setup for all of my experiments. In the future, I believe that we can use the data from the current setting to apply with a general task or scenario. As I focus on an asynchronous task support system, I do not include network latency in this case study.

In this chapter, I investigate the effect of camera latency on a video-see through a handheld device(mobile phone). I conducted two studies: the first study focused on how much different levels of latency can be noticed by the user. The second study investigates how and will latency affect the time and accuracy while performing a high precision task while looking through a handheld device?

3.1 Related Work

MacKenzie et al. defined the definition of latency as ”delay between input action and the output response” [58], this can happen from several factors: a delay of an input device, time to process the data, and time to rendering output. The effects of latency have been studied and explored by many researchers in various settings.

3.1.1 Effect of Latency on Screen Interaction

The latency of a screen interaction is the delay between the time user input and the output present on the screen. There are two types of input in screen interaction; direct and indirect input. Direct input is where the user directly touches and interacts with the screen in the same position where it will show the feedback or output. Indirect input is where the user uses an external device to interact with the system for example using a mouse or pointer.

Many research-related to an indirect input device such as using a mouse for interaction report that the error rates increase significantly if the latency is more than 110 ms [75, 77, 76]. The latency can affect some tasks such as steering, which shows the effect on the performance with only 16 ms latency [30]. For direct input such as touch screen display, Ng et al.[69] studied the effect of latency on dragging actions and found that the user was able to detect the difference in latency levels at only 2.3 ms. For the tapping task, Jota et al. [46] report that the different latency at 63 ms is noticeable to the user. Many researchers also try to overcome the problem of latency [9, 16, 42], some developing a near-zero latency interaction system [23, 46].

These researches have explored and investigate the effect of system input latency or the delay between input and the output showing on display. However, my target is the camera latency or the delay between an action, performing behind the camera, and the display of the action on the screen.

3.1.2 Effect of Latency in Virtual Reality

Several researchers have explored the effect of latency in a virtual reality (VR) environment. Nelson et al.[68] reported on the head-mounted display setting that the latency at 50-100 ms can affect the user's ability to follow a virtual object. Meehan et al.[63] report a similar finding where the effect of latency at 50 and 90 ms is enough to affect the user perception in a virtual environment. Not only on perception but the latency also be reported to affect performance such as task completion time [94].

3.1.3 Effect of Camera Latency on task performing in Video-See Through Display

Most of the studies about the effect of camera latency on video-see through display occur in the medical field. Many research focuses on micro-task such as robotic and laparoscopic surgery. These researches have shown that camera latency affects the operating performance such as increase the operation duration and reduce accuracy [1, 26, 50, 53, 78, 83]. Later studies found that only 105 ms of latency is enough to affect the performance in laparoscopic surgery [53].

These researches are similar to what I want to address, all of the laparoscopic surgery using an indirect input devices such as a robot arm to perform the task. This differs from directly using hands to perform the task, where the latency can affect the user perception because of the movement of hands in a physical world do not match with the movement on the screen. Although these studies provide knowledge to understand the effect of camera latency on a 2D display. However, the effect of camera latency using hands to operate directly on the task is remains unexplored.

3.2 Latency Measurement

The equipment I used for measurement consist of 1) a smartphone (Google Pixel 3) with a screen refresh rate at 60 Hz, 2) a tablet (IPad pro-2018) with a screen refresh rate at 120 Hz, 3) two photo diodes, 4) an amplifier (Figure 15), and 5) a digital oscilloscope (Figure 16). The setup is shown in Figure 17, a smartphone place on a holder faces a back camera down and take a video streaming at a tablet. The tablet screen is showing a switching checkerboard pattern (Black and white), which invert a color every second. Next, two photo diodes connected to an amplifier then connect to a digital oscilloscope. One diode is placed at the top-left corner of the switching checkerboard pattern on a tablet while another placed bottom-left of the video stream checkerboard pattern on a smartphone. The position of diodes was based on the pixel rendering order in each device, this is very important to avoid an unintentional latency as the whole screen cannot be rendered at once. Using an amplifier, each diode signal level jumps to 5 V if the diode is illuminated by light. The digital oscilloscope is used to measure the time elapses between the two signals from a smartphone to tablet. All of the equipment was set up in a light control environment (close room without light source form outside).

I used a custom video camera application in the smartphone with the following setup: The video resolution at 1280 × 720 pixels with camera fps at 60 frames per second(FPS) and fixed camera exposure time to 9.97 ms. To control the latency I modify a camera application to have an image buffer up to 10 frames. By showing old frames I can increase the latency in steps of approx 1/60 of a second. For example, if I wanted to be ”five frames behind” I skipped the first five frames and started to render from the 5th frame onward every frame (Table in Figure 18). For every latency used in the experiment (Figure 18) I took 50 repetitions of latency measurement to eliminate errors caused by synchronisation of sampling and camera exposure time (e.g. a change of checkerboard pattern just after camera finished acquiring frame results in approx 10ms measurement error).

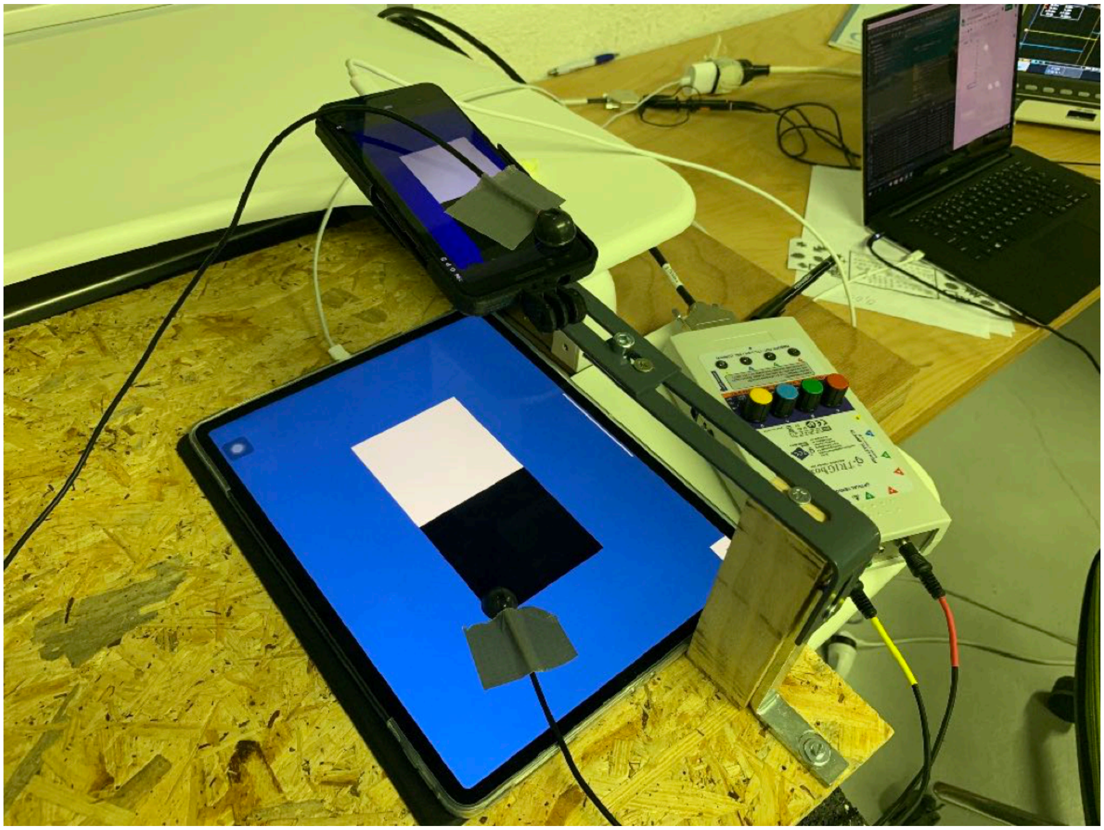


Figure 15: Picture of the system setup for measuring latency which consists of Pixel 3 place on a holder, IPad Pro with black and white checkerboard pattern, two photo diodes,and an amplifier

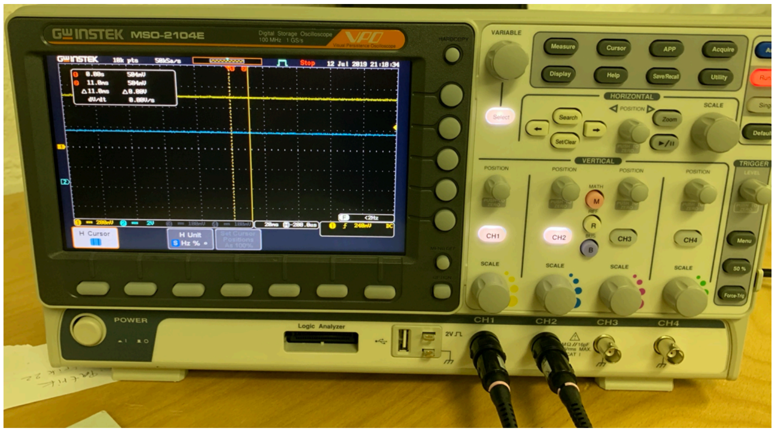


Figure 16: Picture of a digital oscilloscope.

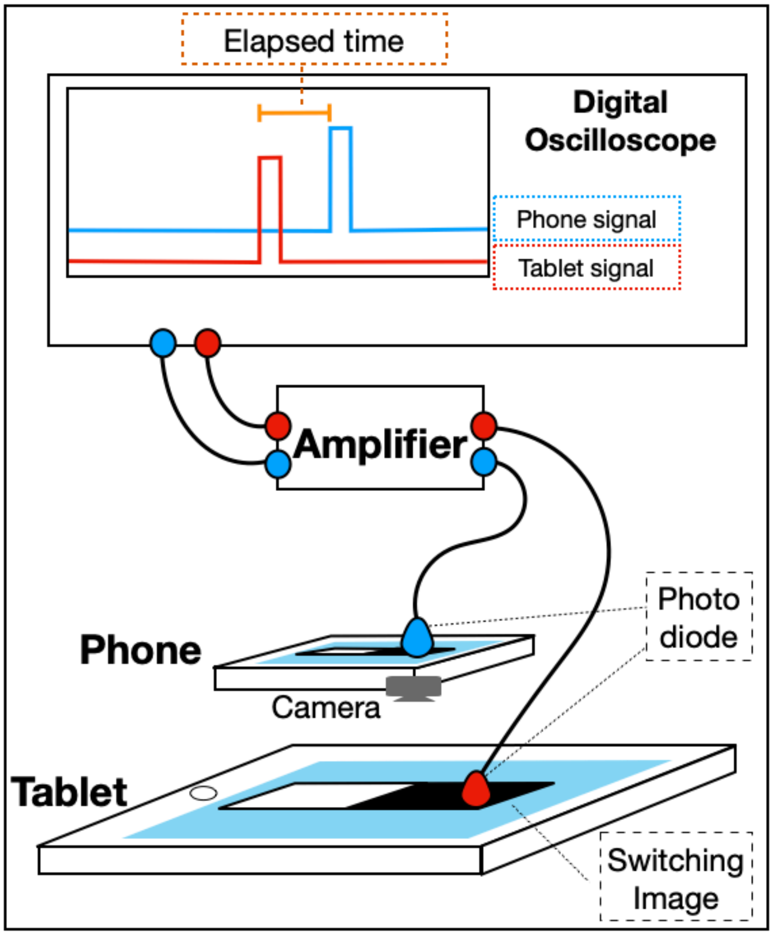
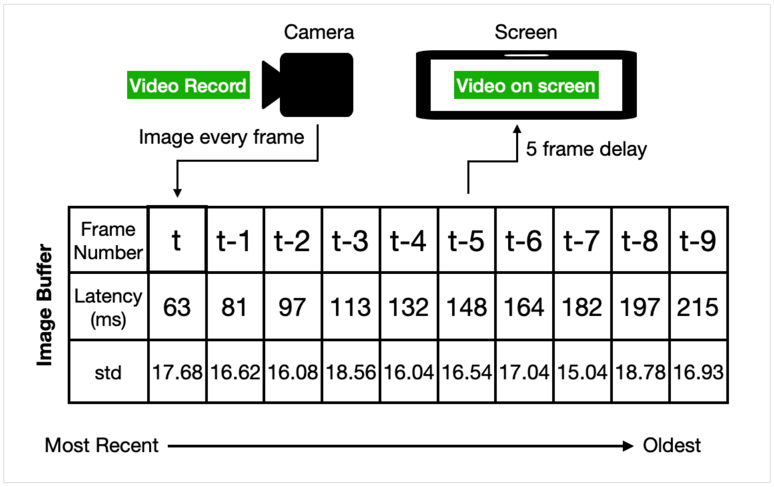


Figure 17: Diagram for the system setup for measuring latency. On the bottom is a tablet computer with a checkerboard pattern on its screen. Above is a smartphone with a camera facing the tablet. Each screen has a photo diode attached. The oscilloscope measures the elapsed time between an event on the tablet screen and the same event shown on the smartphone screen.



Frame Number	t	t-1	t-2	t-3	t-4	t-5	t-6	t-7	t-8	t-9
Latency (ms)	63	81	97	113	132	148	164	182	197	215
std	17.68	16.62	16.08	18.56	16.04	16.54	17.04	15.04	18.78	16.93

Figure 18: With system latency *t* we could increase latency by first skipping n-1 frames and start showing from n-th frame onward from the buffer. For example by skipping first five frames and start showing from the 5th frame on we increased the latency by 148 ms. The bottom line shows that standard deviation from our measurements.

3.3 Study A: Just Noticeable Difference

The objective of this study is to identify which levels of latency participants will notice the difference or Just Noticeable Difference (JND). JND is the smallest change in stimuli that users can detect [23].

3.3.1 Hypothesis

According to the previous work on latency showed JND values range between 5-11 ms [23, 70]. I assume that it should be the same in this study.

- **H1**: Participants will notice the difference at 1 level difference [*t*, *t-1*].

3.3.2 Tasks

To evaluate the JND on high precision tasks such as soldering I chose a pointing task. Participants had to touch the targets (icon/symbol) on a physical rigid half sphere with a 10 cm diameter with a tip of the pen. I printed and placed the targets directly on the object instead of using real augmented reality to avoid the

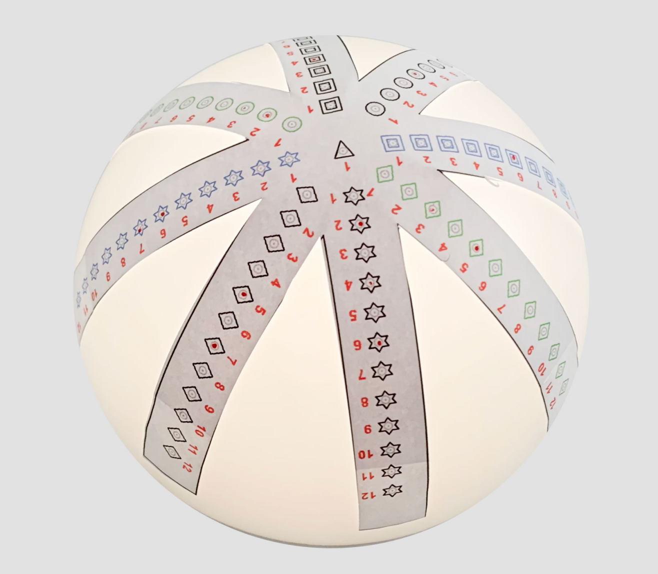


Figure 19: A half sphere with targets. Each target was marked by an empty symbol 3 mm, inner circle 1.2 mm, and an inner dot of 0.34 mm. The targets were numbered from the top (no. 1) towards the edge (no. 12).

effects of AR tracking error and jitter problem. I chose a three dimensional surface as in many remote expert situations one has to work with non-planar surfaces in particular when working with organic materials (e.g. video see-through systems in the medical domain).

The half sphere had 8 groups of 12 targets running from top to the edge of the half sphere in a star shaped manner as shown in Figure 19. Each target was marked by a group specific symbol/icon with a 3 mm diameter, an inner circle of 1.2 mm, and the inner dot of 0.34 mm.

Participants were asked o touch dots within targets as instructed by the experimenter while looking at the half sphere though the phone screen showing a camera view. The phone was placed on the holder but can be freely adjusted to any angle as seen fit by each participant. Participants were also allowed to rotate and move the half sphere but were not allowed to lift it up.

3.3.3 Procedure and Participants

I recruited a total of 3 participants from the local university for this study.

The procedure for one pair of latencies is following these steps:

1. Participants touching 3 point base with the phone A.
2. Participants touch another 3 point with the phone B.
3. Answering following questions:
   - Do you notice any difference between the two latencies?.
   - If yes, which setting do you think have better latency (lower latency).
4. Repeated the comparison 10 times for each latency pair: 4 times the first phone had lower latency, 4 times the second phone had lower latency, and 2 times both phones had the same latency. The order of comparison was randomly assigned in a counterbalanced order.

Participants always compared the lowest latency (*t*) with a higher latency (*t-1*, *t-2*, *t-3* and *t-4*). Participants had to compared four latency pairs ([*t*, *t-1*], [*t*, *t-2*], [*t*, *t-3*], [*t*, *t-4*]).

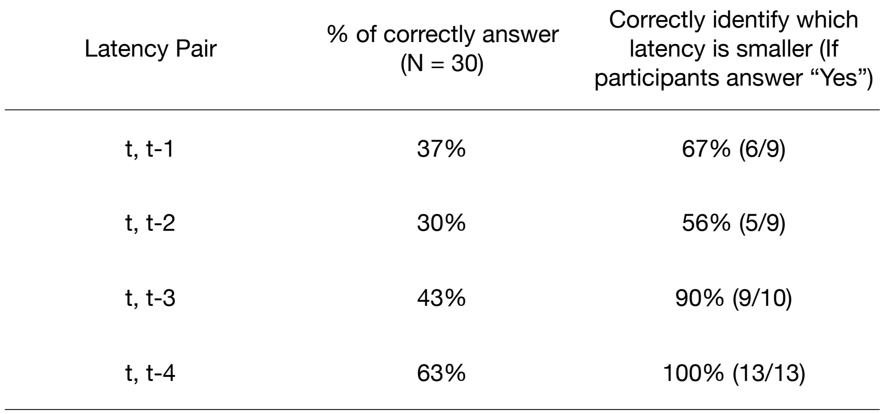
I count the total number of correct answers per 1 comparison. The condition for an answer to be correct is: 1) If participants answer ”No” in fake comparing, 2) If participants were able to tell which setting has low latency in real comparing.

I did not compare higher latency values since I noted in a pre-study that latency *t-5* was already very noticeable; although, they might as well happen in for example the remote assistance scenario when the instructions might come with even higher delays.

3.3.4 Results and Discussion

I calculate the percentage of correct answers in each comparison. I consider the results are noticeable if the correctness is more than 50%. From the results, participants start to notice the difference at pair [*t*, *t-4*]. (132.38 ms) with the correctness at 63%. In the smaller pair, participants rarely able to identify if two latencies are the same or not. Participants able to give the correct answer at 37% on [*t*, *t-1*], 30% on [*t*,*t-2*], and 48% on [*t*,*t-3*] (Table 4). Thus the **H1** was rejected.

According to observation, feedback, and discussion with participants, one of the reasons that participants notice the difference at high latency is the task itself



Latency Pair	% of correctly answer (N = 30)	Correctly identify which latency is smaller (If participants answer “Yes”)
t, t-1	37%	67% (6/9)
t, t-2	30%	56% (5/9)
t, t-3	43%	90% (9/10)
t, t-4	63%	100% (13/13)

Table 4: Identifying differences in latency between two conditions.

does not require speed. All participants focus and concentrate on the pen while performing the task slowly and carefully. Participants start to notice only when they move their hand, pen, or the latency was very high.

3.4 Study B: The effect of latency on user performance

From the results of study A, I found that 132.38 ms (*t-4*) is the latency that most participants able to notice. In this study, I want to identify whether the latency will affect task performance or not.

3.4.1 Hypothesis

I have two hypotheses in this study

- **H1**: Phone with high latency will have lower success rate of task.
- **H2**: Phone with high latency will take longer time to finish task.

3.4.2 Tasks

The task in this study is similar to the previous study, which is touching the targets on the half sphere with 10 cm diameter. In this time, participants have to touch the inner dot (0.34 mm) locate in the middle of the inner circle (size of 0.12 cm) which located in the middle of the icon/symbol (size of 0.3 cm) (Figure

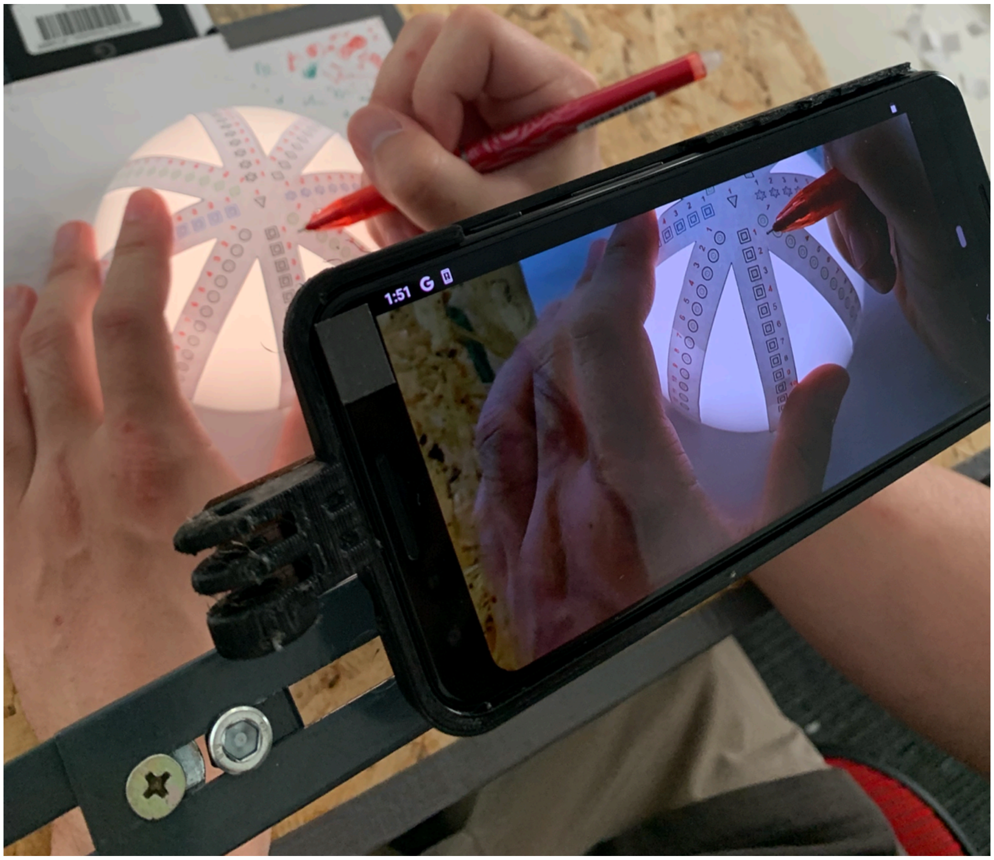


Figure 20: Example of the task: User touching targets on a half sphere with a tip of a pen.

20). After touching one icon, participants have to rotate the half-sphere to left or right and not allow to touch the next target on the same row as the previous one.

Similar to the previous study, the sphere had eight (8) groups of 11 targets running from top (numbered 1) to the edge of the sphere (numbered 11) in a star shaped manner. However, in this study, the targets in each group were divided into three clusters based on difficulty to touch them with a pen tip: 1-4 easy, 5-7 medium, and 8-11 difficult. The difficulty increases by moving from the top of the sphere down. The task involved touching one target within each difficulty cluster in one group of targets (3 touches per group of targets). The number of targets to touch was given by the experimenter. To remove shadows and light noise that might affect task performance, I illuminated the half sphere from within. Again, participants were also allowed to rotate and move the sphere but were not allowed to lift it up.

3.4.3 Procedure and Participants

I recruited a total of 9 participants from the local university (7 males and 2 females; average age: 31.6 years (SD = 7.68); range: 23-44 years).

The experiment took approximately 30-40 minutes per participant. First, a participant spent time up to 10 minutes being tutored (explanation, presentation, and practice level). Participants were instructed to complete the task as fast as and as accurate as possible.

I used 6 conditions: one without a phone and five with phones each with a different latency value. I marked the phones with colours: black (*t*), red (*t-3*), blue (*t-4*), green (*t-5*) and yellow (*t-6*) (Figure 21). I used the *no-phone* condition as a baseline and 3D printed the frame of the size and shape of the phone used in the study. In *no-phone* condition participants also had to cover their non-dominant eye to keep monocular vision constraint across all conditions. The results of Study A showed participants were only able to tell the difference at latency *t-4*, hence I decided to skip latencies *t-1* and *t-2*. The order of conditions was counter balanced for each participant, with *no-phone* and *black-phone* (lowest latency) conditions always following one another.

The procedure for one condition is following these step:

1. Participants were asked to touch three numbers.

2. Participants were asked to check and review the correctness.

3. Repeated step (1)-(2) six times until participants touched a total of 18 targets.

4. After finishing each condition participants were asked to rank the current phone by placing it on the desk from best to worst (left to right).

5. Take a short break (1 - 2 minutes) before move to next condition.

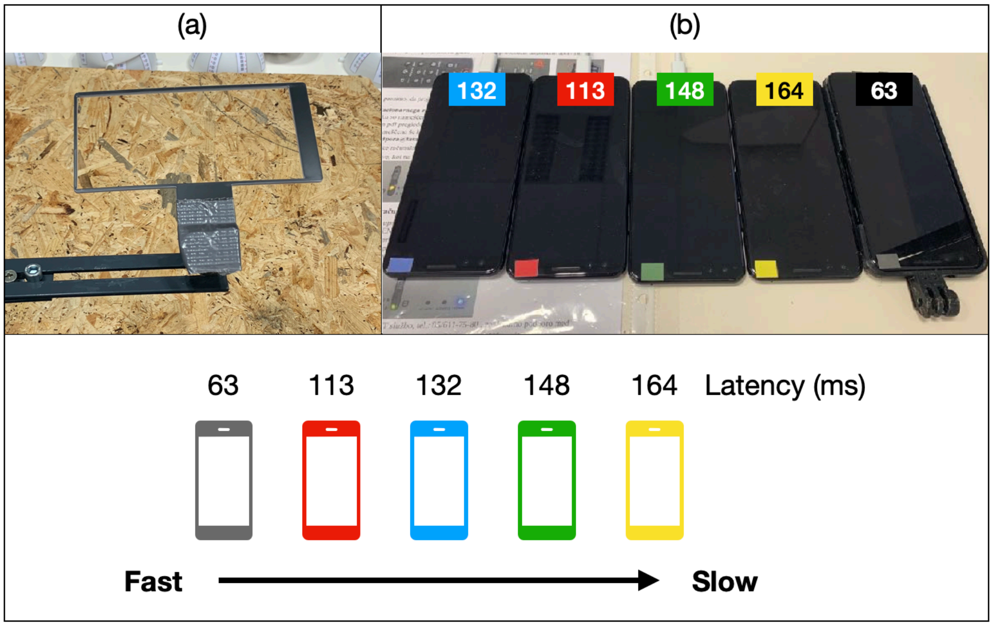


Figure 21: The user had to touch targets on a half sphere with tip of a pen looking through the phone screen.

In case of *no-phone* and *black-phone* condition participants have to filled the NASA-TLX questionnaire.

All data measure in this experiment was collected manually by the experimenter. I recorded the time as a sequence per three touches and total time for each phone. For the correctness, I consider touching with-in inner circle as a success. Touching outside the inner circle or have ink spread outside consider as a failure (Figure 22).

3.4.4 Results

The result is divided into: time and accuracy, phone ranking, and workload assessment (NASA-TLX). The accuracy results were consistent across all conditions: for individual touch events as well as when I consider sequences with 0 or 1 failure. The time results also consistent for all conditions. However, *no-phone* condition seem to have faster performance and smaller standard deviation compared to others (highlighted in Table 5). The ranking results show participants could not correctly rank phone with latency *t-4* or lower but could do this

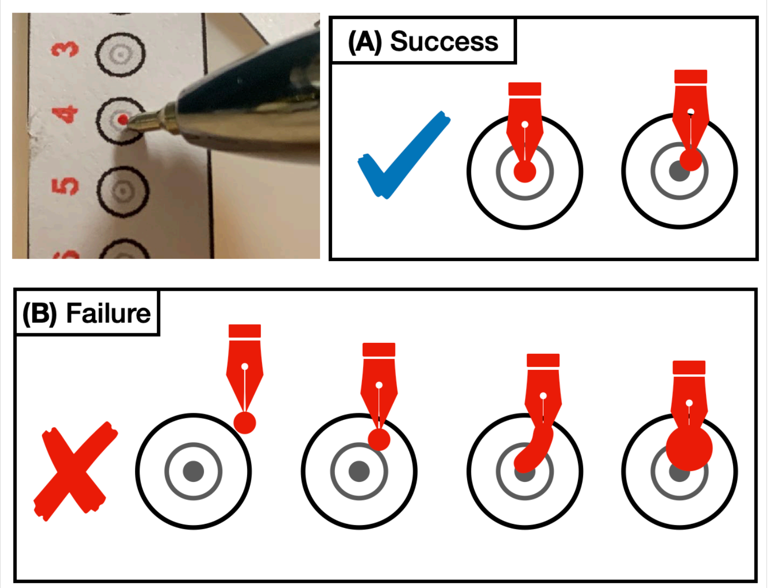


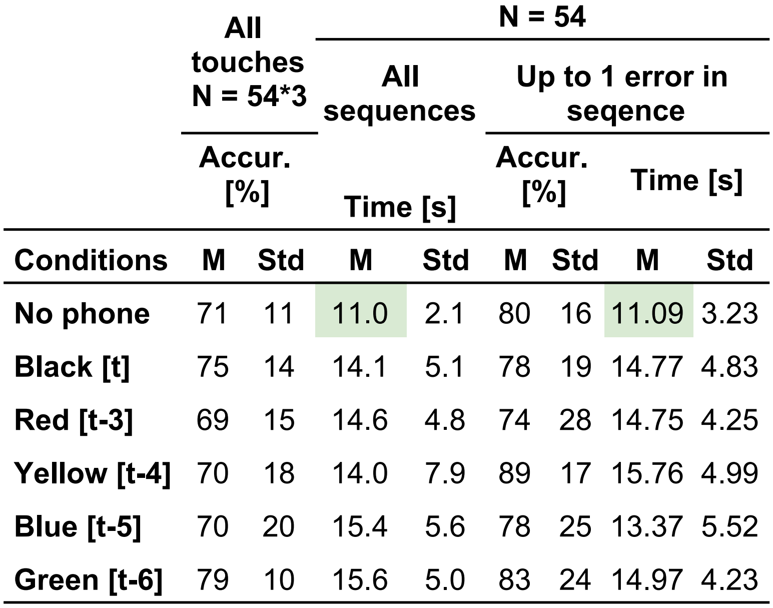
Figure 22: (A) A touch was considered successful if the pen touched the target within the inner circle. (B) All other cases are considered unsuccessful.

for higher latencies (observe median scores in Table 6). The NASA-TLX results show that despite masking one eye to ensure monocular vision participants still found the *no-phone* condition as less workload demanding in all aspects of the NASA-TLX questionnaire. Overall **H1** and **H2** were rejected.

3.4.5 Discussion

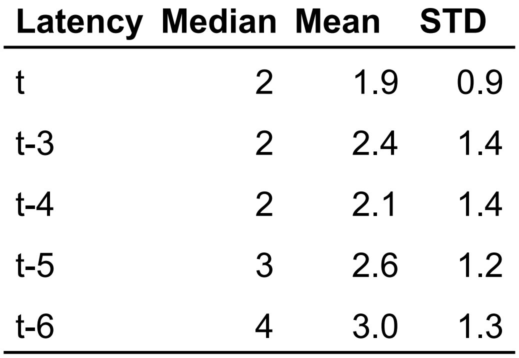
The results reveal that latency does not affect the performance in time and accuracy. Participants focused on the touching task and only noticed the latency when moving or swinging either hand or pen. Some participants performed the task very slowly and did not notice the latency at all. The main finding is that my task was very static and not affected by latency, which led to very similar results across all conditions. The depth perception caused more problems. Participants focused first on accuracy followed by speed, and in some cases, higher latency even caused them to be more cautious.

The ranking results of different latencies showed that user preference starts to manifest from latency *t-4* onward. This suggests that users could mitigate



	All touches N = 54*3		N = 54					
			All sequences		Up to 1 error in seqence			
	Accur. [%]		Time [s]		Accur. [%]		Time [s]	
Conditions	**M**	**Std**	**M**	**Std**	**M**	**Std**	**M**	**Std**
No phone	71	11	11.0	2.1	80	16	11.09	3.23
Black [t]	75	14	14.1	5.1	78	19	14.77	4.83
Red [t-3]	69	15	14.6	4.8	74	28	14.75	4.25
Yellow [t-4]	70	18	14.0	7.9	89	17	15.76	4.99
Blue [t-5]	70	20	15.4	5.6	78	25	13.37	5.52
Green [t-6]	79	10	15.6	5.0	83	24	14.97	4.23

Table 5: Time and accuracy results.



Latency	**Median**	**Mean**	**STD**
t	2	1.9	0.9
t-3	2	2.4	1.4
t-4	2	2.1	1.4
t-5	3	2.6	1.2
t-6	4	3.0	1.3

Table 6: Ranking of different latencies.

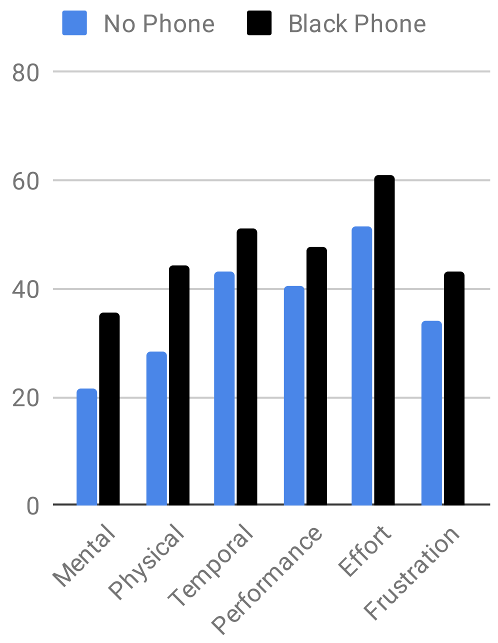


Figure 23: NASA-TLX score.

the effect of latency by simply putting more effort into the task, however, would prefer to use a faster phone where *t-4* is already good enough for the task at hand.

The studied task concentrated on a high precision scenario where it is common to perform slow movements. Furthermore, the scenario was predominantly static (e.g. the phone is on the stand, the augmented object is resting on the table surface). These two facts mean there was not a lot of movement in the scene. This might be one of the reasons why I only saw a limited effect of latency on the task I studied. Previous work on latency showed JND values for more dynamic tasks (e.g. dragging) range between 5-11 ms [23, 70]. This is a magnitude of 5-10 times smaller than values reported for tapping.

3.5 Limitation

There are several limitations to the studies. The task was very static the users can not move the device and the targets are not moving. During the task, participants were asked to perform while looking through the screen only. I do not consider the situation where the participants have a choice to see the physical task directly. I believe that by allow participants to see the physical task directly, they can avoid most of the limitations in a video see-through display especially the depth perception. In addition, to avoid the problem of tracking and rendering problems, I did not use augmented reality as instruction in my studies. With the limitation of hardware and devices, the lowest latency that I can get on the device is 63 ms. The number of participants is low on both studies, the results might be more clear if we manage to recruit more participants.

3.6 Chapter Summary

In this chapter, I focus on monocular video-see through display and investigate the effect of latency on the user while performing a task requiring high precision. I choose the mobile phone as video-see through display device due to the popularity and availability of the platform. First, I developed a mobile phone camera application that can adjust latency up to 10 levels and measure it. Then, I conducted a study to measure JND to see when participants start noticing the difference between the two latency values. Finally, I conducted another study to investigate how does latency affects time and accuracy in high precision tasks.

4. Conclusion

In this thesis, I investigated the human perspective and interaction in handheld augmented reality for an asynchronous task support system. I conducted two studies focus on two areas based on the role of the user in the task assistance scenario: input an AR instruction by an expert user and receive and follow an AR instruction by novice/local user. In the first study, I focus on a 6 DoFs object manipulation technique in HAR utilize the gravity information to assist in orientation control and evaluated with the a-state-of-the-art method. The results showed that SlidAR+ improves performance when the target pose is aligned to gravity. This can be useful in an asynchronous task support system where the expert user has to place an AR content on a man-made artificial environment such as a wall, table, or pillar. In the second study, I investigate the effect of camera latency on task performance while looking through a mobile phone screen. The results of the preliminary studies have shown that camera latency has no effect on user performance on both time and accuracy in a static situation. This information can be useful when we design a HAR task support system in a static situation as we do not have to worry about the camera latency we can focus the resource to address the other limitations in HAR.

4.1 Finding

I list some of my findings and what I have learned during both studies.

4.1.1 Object manipulation in HAR

- **Orientation control using gravity**
  In an optimal situation where we want to place a virtual object align with the gravity direction, our gravity constraint rotation shows promising results. Especially, nowadays where most of the real-world man-made artificial environment always placed aligned to the gravity (parallel or perpendicular to the ground).

- **Initial orientation**
  The initial orientation alone does not show to affect the performance in

object manipulating as it can be rotated to other orientation immediately after that. However, it can be useful when combining it with other rotation constraint techniques.

- **Physical fatigue**
  We found that the size and weight of the device along with that way of holding a device affect the physical effort and fatigue. One hand holding methods does not suitable to operate with a heavy and large device for a long time.

4.1.2 The effect of latency

- **Just Noticeable Difference (JND)**
  In our study, we found JND value at the baseline latency **t**(63 ms) at 69 ms (JND[63ms] = 69ms). Deber el al. measured JND for a tapping task on a screen and repored JND[1ms] = 69 ms and JND[66.7] = 50 ms [23]. The results are similar even these two tasks seem to look very different. This is interesting and can be extended to the future work at highter baseline latency or system.

- **Low movement & high precision task**
  The high precision task with low movement seems to be affected much from latency. To notice the latency a significant amount of movement is required which rarely happens in my high precision task where most of the participants concentrate on accuracy than speed.

- **Depth information**
  Depth perception seems to have a larger effect on a high precision task. As a handheld device is a monocular video-see through systems, the depth information can not be present during the task. Many times during a pilot test, I observed many participants perform the task very quickly when there is a shadow presented in the setup.

4.2 Future Work

Many possible future work directions can be extended from both SlidAR+ and latency studies.

1. **SlidAR+**

    - **Improve interface for larger device**
      At first, I designed an interface and tested it on a smaller device such as IPad Air. I keep that design when I change to a larger device (IPad Pro). As I discuss in Chapter 2, two hands holding device is more suitable in a larger and heavy device. Improving a user interface to support two hands holding can greatly improve manipulability for SlidAR+.
    - **Evaluation on HMD environment**
      The current setting focus on a handheld device. However, it possible to extend it to a head-mounted display device to evaluate and explore the performance and usefulness of our system.
    - **Synchronous task support system**
      The current evaluations have been setting based on an asynchronous task support system as the user needs to move to other viewpoints to use SlidAR for positioning. However, in the situation where the user can not control or change the viewpoint freely, the usability of SlidAR+ is still unclear.
    - **Combine with other rotation method**
      My approaches focus on the condition that a virtual object place aligned to the gravity only. I suggest to find and combine a gravity constraint rotation methods with other intuitive rotation methods for non-gravity alignment case.

2. **Camera latency**

    - **Larger environment**
      The current setting focus on micro or small scale tasks. It might

be worth expanding the scale of the task to a bigger environment such as table scale. As a large environment have more space for a high movement task, the user might be able to observe and notice the latency easier than a small scale task.

- **Dynamic situation**
  The high movement has a strong effect on the ability to notice the latency. The more dynamic situation where action requires dragging or an object is moving can be easily affected by latency than a static situation like in the current setting.

- **Other types of display**
  It might worth studying and investigate the effect of latency on other types of display such as head-mounted display or projector base AR system. There is a lot of work already conduct studies in this field but there are many scenarios that have not been studied and covered. So, I think it worth to continues the work.

Acknowledgements

I would like to give my thanks and appreciation to all individuals who help, support, and advised me during my journal for over five years. First, I would like to express my gratitude to Professor Hirokazu Kato for accepting me to his laboratory and supervised me over five years of Master's and Ph.D. courses, along with many variable comments, advice, and knowledge he gave me. I thank Affiliate Associate Professor Takafumi Taketomi for all his contributions and supervision during my Master's degree. I would like to thank Assistant Professor Alexander Plopski for every knowledge about the experiment and user study he gave me, along with many comments about writing. I thank Associate Professor Christian Sandor for many of his comments and for refining my presentation skill. I would like to thank Assistant Professor Yuichiro Fujimoto for all support and discussion he provides to me during my thesis and journal writing. I would like to thank Associate Professor Masayuki Kanbara for all of his comments about my research. I would like to thank my thesis committee: Professor Kiyoshi Kiyokawa for his questions and comments. I would like to give a special thanks to Matjaž Kljun, Klen Čopič Pucihar for taking care of me during an internship at the HICUP laboratory and for an interesting project we did together. The time I spend with your guys during the internship is one of the greatest moments during my Ph.D. life.

Last but not least, I would like to thank all Interaction Media Design laboratory members for every funny thing we did together. I would like to thank all Thai students at NAIST for providing a good community and support each other. I thank my family for continuing to support me during these five years.

Publications

1. **Varunyu Fuvattanasilp**, Yuichiro Fujimoto, Alexander Plopski, Takafumi Taketomi, Christian Sandor, Masayuki Kanbara, Hirokazu Kato, ”SlidAR+: Gravity-Aware 3D Object Manipulation for Handheld Augmented Reality,” Computers Graphics, Elsevier, Vol.95, No.1, pp.23-35, Apr. 2021, DOI: 10.1016/j.cag.2021.01.005

2. (Poster) **Varunyu Fuvattanasilp**, Matjaž Kljun, Hirokazu Kato, Klen Čopič Pucihar, ”The Effect of Latency on Micro Instructions in Mobile AR,” 22nd International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI 2020), The Association for Computing Machinery Special Interest Group on Computer–Human Interaction (ACM SIGCHI), Online, 20 Oct. 2020, DOI: 10.1145/3406324.3410716

3. Alexander Plopski, **Varunyu Fuvattanasilp**, Jarkko Polvi, Takafumi Taketomi, Christian Sandor, Hirokazu Kato, ”Efficient In-Situ Creation of Augmented Reality Tutorials,” Proceedings of the IEEE International Workshop on Metrology for Industry 4.0 and IoT, pp.7-11, Brescia, Italy, Apr. 2018

4. Takafumi Taketomi, **Varunyu Fuvattanasilp**, Alexander Plopski, Christian Sandor, Hirokazu Kato, ”3D Contents Arrangement in Handheld Augmented Reality Application Based on Gravity Vector,” The First International Workshop on Mixed and Augmented Reality Innovations, pp.1-2, Tasmania, Australia, 2016, Nov. 2016

References

[1] Mehran Anvari, Tim Broderick, Harvey Stein, Trevor Chapman, Moji Ghodoussi, Daniel W Birch, Craig Mckinley, Patrick Trudeau, Sanjeev Dutta, and Charles H Goldsmith. The impact of latency on surgical precision and task completion during robotic-assisted remote telepresence surgery. *Computer Aided Surgery*, 10(2):93–99, 2005.

[2] Ronald T Azuma. A survey of augmented reality. *Presence: Teleoperators & Virtual Environments*, 6(4):355–385, 1997.

[3] Nuernberger B., Ofek E., Benko H., and Wilson A. D. SnapToReality: Aligning Augmented Reality to the Real World. In *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems*, pages 1233–1244, 2016.

[4] Huidong Bai, Lei Gao, J. El-Sana, and M. Billinghurst. Markerless 3D Gesture-based Interaction for Handheld Augmented Reality Interfaces. In *Proceedings of the IEEE International Symposium on Mixed and Augmented Reality*, pages 1–6, Oct 2013.

[5] Huidong Bai, Gun A. Lee, and Mark Billinghurst. Freeze View Touch and Finger Gesture Based Interaction Methods for Handheld Augmented Reality Interfaces. In *Proceedings of the Conference on Image and Vision Computing New Zealand*, pages 126–131, 2012.

[6] Huidong Bai, Gun A. Lee, Mukundan Ramakrishnan, and Mark Billinghurst. 3D Gesture Interaction for Handheld Augmented Reality. In *Proceedings of the SIGGRAPH Asia Mobile Graphics and Interactive Applications*, pages 7:1–7:6, 2014.

[7] Domagoj Baričević, Tobias Höllerer, Pradeep Sen, and Matthew Turk. User-perspective augmented reality magic lens from gradients. In *Proceedings of the 20th ACM Symposium on Virtual Reality Software and Technology*, pages 87–96, 2014.

[8] Martin Bauer, Gerd Kortuem, and Zary Segall. " where are you pointing at?" a study of remote collaboration in a wearable videoconference system. In *Digest of Papers. Third International Symposium on Wearable Computers*, pages 151–158. IEEE, 1999.

[9] François Bérard and Renaud Blanch. Two touch system latency estimators: high accuracy and low overhead. In *Proceedings of the 2013 ACM international conference on Interactive tabletops and surfaces*, pages 241–250, 2013.

[10] Matthias Berning, Daniel Kleinert, Till Riedel, and Michael Beigl. A study of depth perception in hand-held augmented reality using autostereoscopic displays. In *2014 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 93–98. IEEE, 2014.

[11] Mark Billinghurst, Adrian Clark, and Gun Lee. A survey of augmented reality. 2015.

[12] Mark Billinghurst and Hirokazu Kato. Collaborative augmented reality. *Communications of the ACM*, 45(7):64–70, 2002.

[13] Michael Boronowsky, Tom Nicolai, Christoph Schlieder, and Ansgar Schmidt. Winspect: A case study for wearable computing-supported inspection tasks. In *Proceedings Fifth International Symposium on Wearable Computers*, pages 163–164. IEEE, 2001.

[14] Doug Bowman, Ernst Kruijff, Joseph J LaViola Jr, and Ivan P Poupyrev. *3D User Interfaces: Theory and Practice.* Addison Wesley Longman Publishing Co., Inc., 2004.

[15] R. O. Castle and D. W. Murray. Object Recognition and Localization while Tracking and Mapping. In *Proceedings of the IEEE International Symposium on Mixed and Augmented Reality*, pages 179–180, Oct 2009.

[16] Elie Cattan, Amélie Rochet-Capellan, Pascal Perrier, and François Bérard. Reducing latency with a continuous prediction: Effects on users' performance in direct-touch target acquisitions. In *Proceedings of the 2015 Inter-*

national Conference on Interactive Tabletops & Surfaces, pages 205–214, 2015.

[17] Thomas P Caudell and David W Mizell. Augmented Reality: An Application of Heads-Up Display Technology to Manual Manufacturing Processes. In *Proceedings of the Hawaii International Conference on System Sciences*, volume 2, pages 659–669, 1992.

[18] Wendy H. Chun and Tobias Höllerer. Real-time Hand Interaction for Augmented Reality on Mobile Phones. In *Proceedings of the International Conference on Intelligent User Interfaces*, pages 307–314, 2013.

[19] Klen Čopič Pucihar, Paul Coulton, and Jason Alexander. Creating a stereoscopic magic-lens to improve depth perception in handheld augmented reality. In *Proceedings of the 15th MobileHCI*, pages 448–451, 2013.

[20] Klen Čopič Pucihar, Paul Coulton, and Jason Alexander. Evaluating dual-view perceptual issues in handheld augmented reality: device vs. user perspective rendering. In *Proceedings of the 15th ACM on ICMI*, pages 381–388, 2013.

[21] Klen Čopič Pucihar, Paul Coulton, and Jason Alexander. The use of surrounding visual context in handheld ar: device vs. user perspective rendering. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 197–206, 2014.

[22] Dan Curtis, David Mizell, Peter Gruenbaum, and Adam Janin. Several devils in the details: making an ar application work in the airplane factory. In *Proc. Int'l Workshop Augmented Reality*, pages 47–60, 1999.

[23] Jonathan Deber, Ricardo Jota, Clifton Forlines, and Daniel Wigdor. How much faster is fast enough? user perception of latency & latency improvements in direct and indirect touch. In *Proceedings of the 33rd CHI*, pages 1827–1836, 2015.

[24] Arindam Dey, Graeme Jarvis, Christian Sandor, and Gerhard Reitmayr. Tablet versus phone: Depth perception in handheld augmented reality. In

2012 IEEE international symposium on mixed and augmented reality (ISMAR), pages 187–196. IEEE, 2012.

[25] Nathaniel I Durlach and Anne S Mavor. *Virtual reality: scientific and technological challenges*. 1995.

[26] MICHAEL D FABRLZIO, Benjamin R Lee, David Y Chan, Daniel Stoianovici, Thomas W Jarrett, Calvin Yang, and Louis R Kavoussi. Effect of time delay on surgical performance during telesurgical manipulation. *Journal of endourology*, 14(2):133–138, 2000.

[27] Omid Fakourfar, Kevin Ta, Richard Tang, Scott Bateman, and Anthony Tang. Stabilized annotations for mobile remote assistance. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, pages 1548–1560, 2016.

[28] Steven Feiner, Blair Macintyre, and Dorée Seligmann. Knowledge-based augmented reality. *Communications of the ACM*, 36(7):53–62, 1993.

[29] George W Fitzmaurice. Situated information spaces and spatially aware palmtop computers. *Communications of the ACM*, 36(7):39–49, 1993.

[30] Sebastian Friston, Per Karlström, and Anthony Steed. The effects of low latency on pointing and steering tasks. *IEEE transactions on visualization and computer graphics*, 22(5):1605–1615, 2015.

[31] Steffen Gauglitz, Cha Lee, Matthew Turk, and Tobias Höllerer. Integrating the physical environment into mobile remote collaboration. In *Proceedings of the 14th MobileHCI*, pages 241–250, 2012.

[32] Steffen Gauglitz, Benjamin Nuernberger, Matthew Turk, and Tobias Höllerer. In touch with the remote world: remote collaboration with augmented reality drawings and virtual navigation. In *Proceedings of the 20th VRST*, pages 197–205, 2014.

[33] Georg Gerstweiler, Emanuel Vonach, and Hannes Kaufmann. Hymotrack: A mobile ar navigation system for complex indoor environments. *Sensors*, 16(1):17, 2016.

[34] Michael Gervautz and Dieter Schmalstieg. Anywhere interfaces using handheld augmented reality. *Computer*, 45(7):26–31, 2012.

[35] Leo Gombač, Klen Čopič Pucihar, Matjaž Kljun, Paul Coulton, and Jan Grbac. 3d virtual tracing and depth perception problem on mobile ar. In *Proceedings of the 2016 EA CHI*, pages 1849–1856, 2016.

[36] Pavel Gurevich, Joel Lanir, Benjamin Cohen, and Ran Stone. Teleadvisor: a versatile augmented reality tool for remote assistance. In *Proceedings of the CHI*, pages 619–622, 2012.

[37] Mark Hancock, Sheelagh Carpendale, and Andy Cockburn. Shallow-depth 3D Interaction: Design and Evaluation of One-, Two- and Three-touch Techniques. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1147–1156, 2007.

[38] S. Henderson and S. Feiner. Exploring the Benefits of Augmented Reality Documentation for Maintenance and Repair. In *IEEE Transactions on Visualization and Computer Graphics*, volume 17, pages 1355–1368, Oct 2011.

[39] Anders Henrysson, Mark Billinghurst, and Mark Ollila. Virtual Object Manipulation Using a Mobile Phone. In *Proceedings of the International Conference on Augmented Tele-existence*, pages 164–171, 2005.

[40] Anders Henrysson, Joe Marshall, and Mark Billinghurst. Experiments in 3D Interaction for Mobile Phone AR. In *Proceedings of the International Conference on Computer Graphics and Interactive Techniques in Australia and Southeast Asia*, pages 187–194, 2007.

[41] Anders Henrysson, Mark Ollila, and Mark Billinghurst. Mobile phone based ar scene assembly. In *Proceedings of the 4th international conference on Mobile and ubiquitous multimedia*, pages 95–102, 2005.

[42] Niels Henze, Sven Mayer, Huy Viet Le, and Valentin Schwind. Improving software-reduced touchscreen latency. In *Proceedings of the 19th International Conference on Human-Computer Interaction with Mobile Devices and Services*, pages 1–8, 2017.

[43] Wolfgang Hürst and Joris Dekker. Tracking-based Interaction for Object Creation in Mobile Augmented Reality. In *Proceedings of the ACM International Conference on Multimedia*, pages 93–102, 2013.

[44] Wolfgang Hürst and Casper van Wezel. Gesture-based Interaction via Finger Tracking for Mobile Augmented Reality. In *Multimedia Tools and Applications*, volume 62, pages 233–258, 2013.

[45] Duy-Nguyen Ta Huynh, Karthik Raveendran, Yan Xu, Kimberly Spreen, and Blair MacIntyre. Art of Defense: A Collaborative Handheld Augmented Reality Board Game. In *Proceedings of the ACM SIGGRAPH Symposium on Video Games*, pages 135–142, 2009.

[46] Ricardo Jota, Albert Ng, Paul Dietz, and Daniel Wigdor. How fast is fast enough? a study of the effects of latency in direct-touch pointing tasks. In *Proceedings of the sigchi conference on human factors in computing systems*, pages 2291–2300, 2013.

[47] Jinki Jung, Jihye Hong, Sungheon Park, and Hyun S. Yang. Smartphone As an Augmented Reality Authoring Tool via Multi-touch Based 3D Interaction Method. In *Proceedings of the ACM SIGGRAPH International Conference on Virtual-Reality Continuum and Its Applications in Industry*, pages 17–20, 2012.

[48] Anantha R Kancherla, Jannick P Rolland, Donna L Wright, and Grigore Burdea. A novel virtual reality tool for teaching dynamic 3d anatomy. In *International Conference on Computer Vision, Virtual Reality, and Robotics in Medicine*, pages 163–169. Springer, 1995.

[49] Hannes Kaufmann and Dieter Schmalstieg. Mathematics and Geometry Education with Collaborative Augmented Reality. In *Computers & graphics*, volume 27, pages 339–345. Elsevier, 2003.

[50] T Kim, PM Zimmerman, MJ Wade, and CA Weiss. The effect of delayed visual feedback on telerobotic surgery. *Surgical Endoscopy and Other Interventional Techniques*, 19(5):683–686, 2005.

[51] Georg Klein and David Murray. Parallel tracking and mapping on a camera phone. In *2009 8th IEEE International Symposium on Mixed and Augmented Reality*, pages 83–86. IEEE, 2009.

[52] Gudrun Klinker, Oliver Creighton, Allen H Dutoit, Rafael Kobylinski, Christoph Vilsmeier, and B Brugge. Augmented maintenance of powerplants: A prototyping case study of a mobile ar system. In *Proceedings IEEE and ACM international symposium on augmented reality*, pages 124–133. IEEE, 2001.

[53] Asli Kumcu, Lotte Vermeulen, Shirley A Elprama, Pieter Duysburgh, Ljiljana Platiša, Yves Van Nieuwenhove, Nele Van De Winkel, An Jacobs, Jan Van Looy, and Wilfried Philips. Effect of video lag on laparoscopic surgery: correlation between performance and usability at low latencies. *The International Journal of Medical Robotics and Computer Assisted Surgery*, 13(2):e1758, 2017.

[54] Daniel Kurz and Selim Benhimane. Gravity-Aware Handheld Augmented Reality. In *Proceedings of the IEEE International Symposium on Mixed and Augmented Reality*, pages 332–335, 2011.

[55] Daniel Kurz and Selim Benhimane. Handheld Augmented Reality involving Gravity Measurements. In *Computers & Graphics*, volume 36, pages 866 – 883, 2012. Augmented Reality Computer Graphics in China.

[56] Jingbo Liu, Oscar Kin-Chung Au, Hongbo Fu, and Chiew-Lan Tai. Two-Finger Gestures for 6DOF Manipulation of 3D Objects. In *Computer Graphics Forum*, volume 31, pages 2047–2055. Blackwell Publishing Ltd, 2012.

[57] Stephan Lukosch, Mark Billinghurst, Leila Alem, and Kiyoshi Kiyokawa. Collaboration in augmented reality. *Computer Supported Cooperative Work (CSCW)*, 24(6):515–525, 2015.

[58] I Scott MacKenzie and Colin Ware. Lag as a determinant of human performance in interactive systems. In *Proceedings of the INTERACT'93 and*

CHI'93 conference on Human factors in computing systems, pages 488–493, 1993.

[59] A. Martinet, G. Casiez, and L. Grisoni. The Design and Evaluation of 3D Positioning Techniques for Multi-Touch Displays. In *Proceedings of the IEEE Symposium on 3D User Interfaces*, pages 115–118, March 2010.

[60] A. Martinet, G. Casiez, and L. Grisoni. Integrality and Separability of Multitouch Interaction Techniques in 3D Manipulation Tasks. In *IEEE Transactions on Visualization and Computer Graphics*, volume 18, pages 369–380, March 2012.

[61] Fabio Marton, Marcos Balsa Rodriguez, Fabio Bettio, Marco Agus, Alberto Jaspe Villanueva, and Enrico Gobbetti. IsoCam: Interactive Visual Exploration of Massive Cultural Heritage Models on Large Projection Setups. In *Journal on Computing and Cultural Heritage*, volume 7, pages 12:1–12:24, 2014.

[62] Asier Marzo, Benoît Bossavit, and Martin Hachet. Combining Multi-touch Input and Device Movement for 3D Manipulations in Mobile Augmented Reality Environments. In *Proceedings of the ACM Symposium on Spatial User Interaction*, pages 13–16, 2014.

[63] Michael Meehan, Sharif Razzaque, Mary C Whitton, and Frederick P Brooks. Effect of latency on presence in stressful virtual environments. In *IEEE Virtual Reality, 2003. Proceedings.*, pages 141–148. IEEE, 2003.

[64] Paul Milgram and Fumio Kishino. A taxonomy of mixed reality visual displays. *IEICE TRANSACTIONS on Information and Systems*, 77(12):1321–1329, 1994.

[65] David Mizell. Boeing's wire bundle assembly project. *Fundamentals of wearable computers and augmented reality*, 5, 2001.

[66] Mathias Mohring, Christian Lessig, and Oliver Bimber. Video see-through ar on consumer cell-phones. In *Third ieee and acm international symposium on mixed and augmented reality*, pages 252–253. IEEE, 2004.

[67] Annette Mossel, Benjamin Venditti, and Hannes Kaufmann. 3DTouch and HOMER-S: Intuitive Manipulation Techniques for One-handed Handheld Augmented Reality. In *Proceedings of the Virtual Reality International Conference: Laval Virtual*, pages 12:1–12:10, 2013.

[68] W Todd Nelson, Merry M Roe, Robert S Bolia, and Rebecca M Morley. Assessing simulator sickness in a see-through hmd: Effects of time delay, time on task, and task complexity. Technical report, AIR FORCE RESEARCH LAB WRIGHT-PATTERSON AFB OH, 2000.

[69] Albert Ng, Julian Lepinski, Daniel Wigdor, Steven Sanders, and Paul Dietz. Designing for low-latency direct-touch input. In *Proceedings of the 25th annual ACM symposium on User interface software and technology*, pages 453–464, 2012.

[70] J.Y.H. Ng, M. Hausknecht, S. Vijayanarasimhan, O. Vinyals, R. Monga, and G. Toderici. Beyond short snippets: Deep networks for video classification. Technical Report arXiv:1503.08909, Cornell University, 2015.

[71] Christopher Nguan, Brian Miller, Rajni Patel, Patrick PW Luke, and Christopher M Schlachta. Pre-clinical remote telesurgery trial of a da vinci telesurgery prototype. *The International Journal of Medical Robotics and Computer Assisted Surgery*, 4(4):304–309, 2008.

[72] Jennifer J Ockerman and Amy R Pritchett. Preliminary investigation of wearable computers for task guidance in aircraft inspection. In *Digest of Papers. Second International Symposium on Wearable Computers (Cat. No. 98EX215)*, pages 33–40. IEEE, 1998.

[73] T. Olsson and M. Salo. Online User Survey on Current Mobile Augmented Reality Applications. In *Proceedings of the IEEE International Symposium on Mixed and Augmented Reality*, pages 75–84, Oct 2011.

[74] Jason Orlosky, Kiyoshi Kiyokawa, and Haruo Takemura. Dynamic Text Management for See-through Wearable and Heads-up Display Systems. In *Proceedings of the International Conference on Intelligent User Interfaces*, pages 363–370, 2013.

[75] Andriy Pavlovych and Carl Gutwin. Assessing target acquisition and tracking performance for complex moving targets in the presence of latency and jitter. In *Proceedings of Graphics Interface 2012*, pages 109–116. Citeseer, 2012.

[76] Andriy Pavlovych and Wolfgang Stuerzlinger. The tradeoff between spatial jitter and latency in pointing tasks. In *Proceedings of the 1st ACM SIGCHI symposium on Engineering interactive computing systems*, pages 187–196, 2009.

[77] Andriy Pavlovych and Wolfgang Stuerzlinger. Target following performance in the presence of latency, jitter, and signal dropouts. In *Graphics Interface*, volume 2011, pages 33–40, 2011.

[78] Manuela Perez, Frederic Quiaios, Pierre Andrivon, Damien Husson, Michel Dufaut, Jacques Felblinger, and Jacques Hubert. Paradigms and experimental set-up for the determination of the acceptable delay in telesurgery. In *2007 29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 453–456. IEEE, 2007.

[79] Juri Platonov, Hauke Heibel, Peter Meier, and Bert Grollmann. A mobile markerless ar system for maintenance and repair. In *2006 IEEE/ACM International Symposium on Mixed and Augmented Reality*, pages 105–108. IEEE, 2006.

[80] J. Polvi, T. Taketomi, A. Moteki, T. Yoshitake, T. Fukuoka, G. Yamamoto, C. Sandor, and H. Kato. Handheld Guides in Inspection Tasks: Augmented Reality vs. Picture. In *IEEE Transactions on Visualization and Computer Graphics*, volume 24, pages 2118–2128, July 2018.

[81] Jarkko Polvi, Takafumi Taketomi, Atsunori Moteki, Toshiyuki Yoshitake, Toshiyuki Fukuoka, Goshiro Yamamoto, Christian Sandor, and Hirokazu Kato. Handheld guides in inspection tasks: augmented reality versus picture. *IEEE Transactions on Visualization and Computer Graphics*, 24(7):2118–2128, 2018.

[82] Jarkko Polvi, Takafumi Taketomi, Goshiro Yamamoto, Arindam Dey, Christian Sandor, and Hirokazu Kato. SlidAR: A 3D Positioning Method for SLAM-based Handheld Augmented Reality. In *Computers and Graphics*, volume 55, pages 33–43, 2016.

[83] R Rayman, K Croome, N Galbraith, R McClure, R Morady, S Peterson, S Smith, V Subotic, A Van Wynsberghe, and S Primak. Long-distance robotic telesurgery: a feasibility study for care in remote environments. *The international journal of medical robotics and computer assisted surgery*, 2(3):216–224, 2006.

[84] G. Reitmayr, E. Eade, and T. W. Drummond. Semi-automatic Annotations in Unknown Environments. In *Proceedings of the IEEE and ACM International Symposium on Mixed and Augmented Reality*, pages 67–70, Nov 2007.

[85] Jun Rekimoto. The magnifying glass approach to augmented reality systems. In *International Conference on Artificial Reality and Tele-Existence*, volume 95, pages 123–132, 1995.

[86] Jun Rekimoto. Navicam: A magnifying glass approach to augmented reality. *Presence: Teleoperators & Virtual Environments*, 6(4):399–412, 1997.

[87] Jun Rekimoto and Katashi Nagao. The world through the computer: Computer augmented interaction with real world environments. In *Proceedings of the 8th annual ACM symposium on User interface and software technology*, pages 29–36, 1995.

[88] Élisabeth Rousset, François Bérard, and Michaël Ortega. Two-finger 3D Rotations for Novice Users: Surjective and Integral Interactions. In *Proceedings of the International Working Conference on Advanced Visual Interfaces*, pages 217–224, 2014.

[89] Nóbrega Rui and Correia Nuno. Magnetic Augmented Reality: Virtual Objects in Your Space. In *Proceedings of the International Working Conference on Advanced Visual Interfaces*, pages 332–335, 2012.

[90] Marc Ericson C. Santos, Takafumi Taketomi, Christian Sandor, Jarkko Polvi, Goshiro Yamamoto, and Hirokazu Kato. A Usability Scale for Handheld Augmented Reality. In *Proceedings of the ACM Symposium on Virtual Reality Software and Technology*, pages 167–176, 2014.

[91] Brett E Shelton and Nicholas R Hedley. Using augmented reality for teaching earth-sun relationships to undergraduate geography students. In *The First IEEE International Workshop Agumented Reality Toolkit,*, pages 8–pp. IEEE, 2002.

[92] Ken Shoemake. Arcball Rotation Control. In *Graphics Gems*, pages 175–192, 1994.

[93] Jeffrey H Shuhaiber. Augmented Reality in Surgery. In *Archives of Surgery*, volume 139, pages 170–174. American Medical Association, 2004.

[94] Richard HY So and German KM Chung. Sensory motor responses in virtual environments: Studying the effects of image latencies for target-directed hand movement. In *2005 IEEE Engineering in Medicine and Biology 27th Annual Conference*, pages 5006–5008. IEEE, 2006.

[95] Ivan E Sutherland. A head-mounted three dimensional display. In *Proceedings of the December 9-11, 1968, fall joint computer conference, part I*, pages 757–764, 1968.

[96] Matthew Tait and Mark Billinghurst. The effect of view independence in a collaborative ar system. *Computer Supported Cooperative Work (CSCW)*, 24(6):563–589, 2015.

[97] Takafumi Taketomi, Hideaki Uchiyama, and Sei Ikeda. Visual SLAM Algorithms: A Survey from 2010 to 2016. In *IPSJ Transactions on Computer Vision and Applications*, volume 9, Feb 2017.

[98] Arthur Tang, Charles Owen, Frank Biocca, and Weimin Mou. Comparative effectiveness of augmented reality in object assembly. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 73–80, 2003.

[99] Can Telkenaroglu and Tolga Capin. Dual-Finger 3D Interaction Techniques for Mobile Devices. In *Personal and Ubiquitous Computing*, volume 17, pages 1551–1572, 2013.

[100] P Thomas and WM David. Augmented reality: An application of heads-up display technology to manual manufacturing processes. In *Hawaii International Conference on System Sciences*, pages 659–669, 1992.

[101] Mihran Tuceryan, Douglas S. Greer, Ross T. Whitaker, David E. Breen, Chris Crampton, Eric Rose, and Klaus H Ahlers. Calibration requirements and procedures for a monitor-based augmented reality system. *IEEE Transactions on Visualization and Computer Graphics*, 1(3):255–273, 1995.

[102] Matthew Uyttendaele, Antonio Criminisi, Sing Bing Kang, Simon Winder, Richard Szeliski, and Richard Hartley. Image-based interactive exploration of real-world environments. *IEEE Computer Graphics and Applications*, 24(3):52–63, 2004.

[103] Daniel Wagner and Dieter Schmalstieg. History and future of tracking for mobile phone augmented reality. In *2009 International Symposium on Ubiquitous Virtual Reality*, pages 7–10. IEEE, 2009.

[104] Daniel Wigdor, Sarah Williams, Michael Cronin, Robert Levy, Katie White, Maxim Mazeev, and Hrvoje Benko. Ripples: utilizing per-contact visualizations to improve user interaction with touch displays. In *Proceedings of the 22nd annual ACM symposium on User interface software and technology*, pages 3–12, 2009.

[105] Shahrouz Yousefi, Farid Abedan Kondori, and Haibo Li. Experiencing Real 3D Gestural Interaction with Mobile Devices. In *Pattern Recognition Letters*, volume 34, pages 912 – 921, 2013.