Computers & Graphics 97 (2021) 42–53

Contents lists available at ScienceDirect

Computers & Graphics

journal homepage: www.elsevier.com/locate/cag

Special Section on VRCAI

Bringing full-featured mobile phone interaction into virtual reality

Huidong Bai[a,*], Li Zhang[b], Jing Yang[c], Mark Billinghurst[a]

[a] Auckland Bioengineering Institute, The University of Auckland, Auckland 1010, New Zealand
[b] Cyber-Physical Interaction Lab, Northwestern Polytechnical University, Xi'an 710072, China
[c] Department of Computer Science, ETH Zurich, Zurich 8092, Switzerland

ARTICLE INFO

Article history:
Received 20 July 2020
Revised 31 March 2021
Accepted 8 April 2021
Available online 18 April 2021

Keywords:
Virtual reality
Augmented virtuality
Head-mounted displays
Smartphone interaction
Text input

ABSTRACT

Virtual Reality (VR) Head-Mounted Display (HMD) technology immerses a user in a computer generated virtual environment. However, a VR HMD also blocks the users' view of their physical surroundings, and so prevents them from using their mobile phones in a natural manner. In this paper, we present a novel Augmented Virtuality (AV) interface that enables people to naturally interact with a mobile phone in real time in a virtual environment. The system allows the user to wear a VR HMD while seeing his/her 3D hands captured by a depth sensor and rendered in different styles, and enables the user to operate a virtual mobile phone aligned with their real phone. We conducted a formal user study to compare the AV interface with physical touch interaction on user experience in five mobile applications. Participants reported that our system brought the real mobile phone into the virtual world. Unfortunately, the experiment results indicated that using a phone with our AV interfaces in VR was more difficult than the regular smartphone touch interaction, with increased workload and lower system usability, especially for a typing task. We ran a follow-up study to compare different hand visualizations for text typing using the AV interface. Participants felt that a skin-colored hand visualization method provided better usability and immersiveness than other hand rendering styles.

1. Introduction

This research explores how a real smartphone can be used inside a Virtual Reality (VR) environment. Typical VR systems use a Head-Mounted Display (HMD) to immerse users into an entirely Virtual Environment (VE). However, VR HMDs separate users from the real world and their actual bodies. The difficulty for the user to perceive tangible objects around themselves inhibits access to common input devices, such as a keyboard or a mouse.

Smartphones with touch screen interfaces have become one of the most popular consumer devices, and there is a strong desire to use them inside VR. For example, people may want to receive calls, check mobile applications, or send text messages while in a VR HMD. However, it is currently not feasible for a VR user to easily operate a mobile phone without taking off the VR headset. In this paper, we present a VR system that enables users to naturally interact with a real smartphone while experiencing virtual content in a VR headset.

A few VR HMDs can be used while interacting with a mobile phone. For example, the HTC Vive Cosmo[1], Windows Mixed Reality,[2] and Sony PSVR[3] headsets have a flip-up design, which allows the user to quickly swap between the VE and reality by flipping the face panel with little effort. The HTC Vive Focus[4] and HoloSwitch[5] enables users to receive calls or messages on their phones by displaying pop-up notifications in the VR HMD. However, these approaches do not share the phone screen's content, and the users could not answer the phone call or reply to the message in any form in the VR environment.

Some researchers used a real-virtual bridge mechanism to create two-way interaction techniques to connect real and virtual objects. For example, Takashina et al. [1] used a quasi-touch interface to capture an actual phone screen and transfer it into a virtual smartphone in VR, and then mapped touch operations in the virtual world back to the real smartphone. However, the user still needed to use the VR controllers as virtual hands to press the virtual touchscreen of the proxy phone. The system provided a better embodiment experience for mobile phone interaction in VR, but its input method was far different from the normal phone operation and limited the user's VR engagement.

* Corresponding author.
E-mail address: huidong.bai@auckland.ac.nz (H. Bai).

[1] https://www.vive.com/ca/product/vive-cosmos/features/
[2] https://www.microsoft.com/en-us/windows/windows-mixed-reality/
[3] https://www.playstation.com/en-us/explore/playstation-vr/
[4] https://enterprise.vive.com/ca/product/vive-focus/
[5] https://www.holoswitch.com/

https://doi.org/10.1016/j.cag.2021.04.004

Fig. 1. A virtual smartphone with real-time screen updating is reproduced in a VR environment. The user can see his/her own hands and touch the virtual phone with tangible feedback.

Augmented Virtuality (AV) merges real-world objects into virtual worlds. Milgram et al. defined Augmented Virtuality as an immersive graphic environment in which some element of reality has been added [2]. For example, an AV interface would be a VR experience that includes a virtual TV with a live camera feed from the user's environment shown on the virtual screen. Similarly, we want to add a virtual phone to the user's VR experience, but show the actual phone screen on the virtual phone, and enable users to see their real hands in VR simultaneously.

In this paper, we use the AV technology to add a virtual smartphone and its operation into the VR system by combing spatial tracking of a real mobile phone, wireless smartphone screen mirroring, and spatial hand detection. This novel interface enables the user to perform touch input on the mobile screen while being fully immersed within the virtual space, as shown in Fig. 1. The user can read or reply to text messages, make or answer phone calls, and browse the image gallery or social media updates by holding the real phone in their hands and touching its screen, without taking off their VR headset.

Our prototype is one of the first systems that brings a real smartphone with natural touchscreen interaction methods into a VR environment. Compared with previous work, our research makes the following contributions:

- We designed and developed an AV system that enables users to freely interact with a fully-featured smartphone through a mobile touch screen in a VR environment while wearing a VR HMD;
- We designed and implemented augmented hand visualization interfaces to improve the on-screen content interaction, especially the text typing, for the mobile phone in AV;
- We conducted a formal user study to evaluate the usability of the AV interface for mobile touch-based interaction in a virtual environment by comparing with the direct touch interaction in reality using the mobile device;
- We conducted a follow-up user study to compare the performance and usability of the interfaces with three different hand rendering styles for mobile typing in an AV application.

In the rest of the paper, we first review related work by comparing them with our approach, and then describe our system's design and implementation. We present two user studies with performance and usability results. We discuss the limitations of our proof-of-concept prototype and provide a conclusion and directions for future work.

2. Related work

Our research mainly investigates operating smartphones (specifically text typing) and hand representation in VR. In this section, we review related work in each of these areas and discuss the differences from our system.

2.1. Using smartphones in VR

Smartphones are widely used for mobile Augmented Reality (AR) because of technology advances in sensors and processors, which also supports multimodal screen touch interactions with virtual content (e.g., pointing [3] and manipulating [4]). Some affordable VR HMDs (e.g., Google Cardboard[6] and Samsung Gear VR[7]) use the smartphone as a display, but the user cannot touch it with their hands while viewing the virtual content, and so lose the benefit of rich touchscreen interaction. In this research, we aim to bring the smartphone into VR to access all of the phone features instead of only viewing the VR content.

In earlier research, smartphones with wireless connection, keypad, touch screen, and gyroscope sensing were mainly used for various inputs in VR applications. For example, Larsen et al. [5] connected a cellular phone to a VR system so that the user could press the arrow keys to move the virtual objects shown on the phone screen. Liang et al. [6] used a mobile phone to provide movement and tilt input as an alternative to VR controllers for navigation. Steed and Julier [7] used a smartphone to support portable VR content output and screen-touch input with virtual objects. However, most basic interactions using mobile phones have been replaced by controllers [8], except in some specific applications (e.g., 3D modeling [9], or spatial design [10]).

Some smartphones have been partly integrated into commercial VR systems to enable viewing call notifications or text messages in the VR headset, such as the HTC Vive Focus or HoloSwitch. The smartphone itself was still blocked outside of the virtual world, and the screen cannot be directly viewed in these VR systems. Alaee et al. [11] captured the phone appearance, screen, and the user hands with a depth camera mounted in front of a VR headset and rendered everything as an immersive point-cloud in VR. However, the point-cloud based phone screen rendering had a low resolution, limiting precise operations on the phone like text input and character selection.

In contrast, some researchers tracked the mobile phone and rendered it as a virtual proxy with the corresponding screen content, and made touch interfaces partially available. For example, Chang et al. [12] used a Leap Motion[8] and an IMU sensor to track a physical smartphone and rendered its virtual replica with the same appearance in VR. Their system supported photography of the VR scene using intuitive camera operations on the virtual smartphone. However, the user could not see his/her own hands while holding the virtual phone, and the photo-taking was not as natural as in the real world. Similarly, Desai et al. [13] used an image-based approach to track the hands and the smartphone with an additional camera, and then augmented the content onto the corresponding VR region. The system could open a virtual window for users to see the real world and use the physical smartphone directly. However, it only provided a 2D camera feed for the phone operation in VR, and they did not perform an evaluation of their method.

2.2. Hand representation in VR

Representing hands in the VE to improve natural interaction and presence [14] has always been an important research topic. Hand pose can be detected with various sensors and projected into VR systems. For example, the Leap Motion controller is one of the widely used sensors to track hand movements, and researchers

[6] https://arvr.google.com/cardboard/
[7] https://www.samsung.com/global/galaxy/gear-vr/
[8] https://www.ultraleap.com/product/leap-motion-controller/

have attached it to the front panel of VR headsets for spatial gesture input [15]. Some VR controllers were also capable of roughly positioning virtual hands and sensing finger movements based on the button pressing, such as the Oculus Touch. The hands could also be detected with colored [16] or retroreflective [17] markers attached.

However, most of these methods were not always efficient especially when the hands were partially occluded or holding objects. Given this limitation, a dense 3D point-cloud method was preferred to generate the hand model [18] and support natural 3D interaction [19] in VR without relying on pose estimation. Together with color-based foreground segmentation, better hand recognition could also be provided [20].

With advanced graphic processing technology, virtual hands can be rendered in different styles (e.g., hand skeletons, fingertips, cartoon-like or realistically textured palms). Grubert et al. [21] found that different hand representations could influence error rate and presence for typing in VR while using a standard physical keyboard. Schwind et al. [22] showed that realistically rendered hands helped the user obtain a higher level of presence. Using head-mounted depth cameras collocated with the VR headset, users could reproduce their body parts as 3D point-cloud in VR [23]. User studies found that point-cloud virtual hands improved intuitiveness and self-perception when interacting with VR objects [24], and were more preferred due to better performance, less distraction, and a higher sense of presence [25]. In addition, virtual hands can also be rendered transparent for users to see occluded objects behind the VR hands. For example, Buchmann et al. [26] made hands and tools selectively transparent to enhance the human perception of the virtual surroundings. Knierim et al. [17] made virtual hands semi-transparent while typing on a keyboard in VR, which improved performance for inexperienced users compared to opaque rendering styles.

In our work, we use an RGB-Depth camera to detect the user's hands in a 3D point-cloud format and then bring them into our VR system. We also explore how different hand visualization styles can enhance user performance and experiences while using a mobile phone in VR.

2.3. Text input on mobile phones in VR

For general text input on computers, a keyboard would be the first choice by default. Prior research showed that simple virtual keyboards had lower input rates than physical keyboards due to the lack of haptic feedback [27]. Some researchers have introduced physical keyboards into VR to provide easy text input solutions [28]. However, these methods were not applicable for smartphones since the mobile virtual keys on the mobile screen were too small for users to recognize with a limited resolution of the HMD, which leads to inefficiency and frequent typos.

Alternative methods have been applied for smartphone text input in VR. Boustila et al. [29] used a confirm-on-release typing paradigm to interpret the tap point with a circle on the smartphone keyboard and enabled typing in VR. HoVR-Type [30] used a hovering function to visualize the finger point while typing on the virtual phone screen. However, with these systems the user could not see his/her hands in VR, so the performance heavily relied on familiarity with the typing interface.

Supporting virtual hand visualization while typing is critical to guarantee a lower text entry error rate [31], but it can be counterproductive in some cases. Lin et al. [32] visualized the hands and keyboard to improve the immersive VR experience. However, their user study found that the keyboard and fingertips were occluded by the hand mesh, which decreased the first key accuracy. Therefore, we choose to use different hand rendering styles and explore how they can affect text typing performance on smartphones in VR.

(a) (b)

Fig. 2. (a) System setup; (b) Calibration between the attached camera and the VR system.

In summary, most previous works have not successfully introduced full-featured smartphone interaction into VR, and text typing has not been well addressed with the current controller options. We developed our novel system to overcome the above problems and create a compelling solution.

3. System design

We used an Oculus Rift S[9] as our VR HMD in our system since its high-resolution display (1280×1440 at 80Hz) provides a relatively clearer view of the virtual screen content. Moreover, Oculus Rift S uses built-in cameras to capture and translate the user's head and controller movements into VR no matter which way he/she is facing and provides room-scale tracking without external sensors. Our system also enables the user to see his/her 3D digitized hands and the smartphone in a virtual environment. Both virtual hands and the phone are correctly aligned with the physical hands and phone, duplicating the mobile phone interaction with very low latency. Thus, users can answer phone calls or send messages without taking off their VR headsets. The virtual hands can also be rendered in three different visual styles to affect the user experience and performance. We developed our prototype system by combining four main technologies; (1) 6 Degree-of-Freedom (DoF) phone tracking, (2) Phone screen live mirroring, (3) Hand segmentation with depth sensing, and (4) Hand visualization with graphics shaders.

3.1. Configuration and calibration

We set up the system in an office room. As shown in Fig. 2a, the user can sit on a chair or stand while wearing the Oculus headset tethered to a VR-ready computer for content rendering. To capture the user's hands, an RGB-Depth sensor was attached horizontally on the headset's front panel and connected to the same computer via the standard USB cable. The vertical field of view (FoV) of the depth camera (41.5°) was smaller than that of the HMD (110°), so we adjusted the depth camera to be vertically tilted down 16° with a spacer so that the hands could be observed more easily. The user can hold a smartphone with no cables connected but with a customized Oculus controller attached to the top edge for accurate 6DoF tracking by the Oculus headset. Both the VR computer and the mobile phone were connected to the same local wireless network to ensure fast data exchange. The mobile phone can then wirelessly cast its screen to the computer at its native resolution

[9] https://www.oculus.com/rift-s/

(a) (b)

Fig. 3. (a) The modified tracker is installed on the top of the smartphone; (b) A new battery is attached to the bottom of the phone for better weight distribution.

with negligible delay. A plugin will receive the mirrored screen content and project frames into the VR scene as a live texture on the virtual proxy phone screen.

The RGB-Depth sensor captures the hand in a point-cloud format in its own camera coordinate system (CCS), and then the VR system renders these data points in the HMD coordinate system (HCS). We conducted a calibration between the depth sensor and the VR system to get the transforming matrix (from CCS to HCS) to correctly display the point-cloud hands in the VR scene. As shown in Fig. 2b, a square base with a small notch and an internal protrusion was placed on the left side, holding one controller as the HCS reference; An image marker from the Vuforia tracking library[10] was attached on the right side, working as the CCS reference. Thus, the origin of the marker's coordinate system and its scale were rigidly aligned with the controller's virtual center, both of which can be used as paired sampling points by the marker-based tracking and controller-based tracking, respectively. We captured 30 pairs of position values while moving the headset and looking towards the reference set. We then retrieved the transforming matrix between the two coordinate systems by applying the Iterative Closest Point (ICP) method.

3.2. 6DoF phone tracking

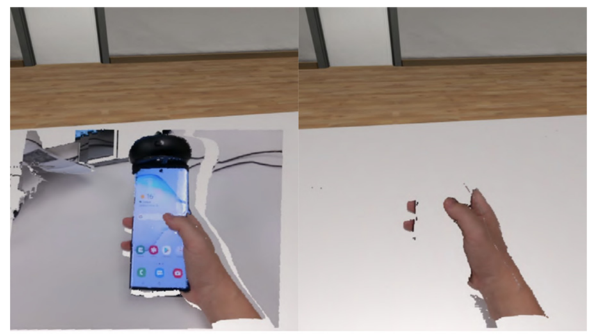
As we mentioned above, the Oculus system has a robust and accurate off-the-shelf visual tracking technique, and its controllers provide stable and fast 6DoF tracking for spatial input. We disassembled and modified the right controller into a portable tracker. The control chip and the trackable ring embedded with infrared LEDs were attached to the top of the phone using a 3D printed connecting case, avoiding any occlusion of the phone screen (Fig. 3a). The controller's virtual center and the phone screen's top center were accurately aligned as one rigid object, and the rendered proxy phone can be placed and tracked in the virtual scene based on the modified controller's position and orientation. The battery cartridge of the right controller was also removed, and an extra new battery was then installed at the bottom of the phone to provide a better overall weight distribution (Fig. 3b). Thus, the user can comfortably hold the device for a long period of time while doing the touch interaction.

3.3. Real-time phone screen mirroring

Once the real phone was successfully tracked and aligned with the virtual proxy phone in VR, we captured the mobile screen's content and fed it into the virtual phone's screen as a live texture. The Scrcpy screen mirroring software[11] was used to wirelessly cast our mobile phone screen content (Fig. 4a) to the VR computer. A customized screen recorder received this content and then streamed it into the VR system and textured it onto the phone surface in real time (Fig. 4b).

(a) (b)

Fig. 4. (a) The tracked phone in reality. (b) The real-time mirrored and textured virtual proxy phone.



(a) (b)

Fig. 5. (a) The aligned point-cloud data in the virtual environment; (b) Hand segmentation from the background.

We established a Wi-Fi 6 wireless connection between the mobile phone and the VR computer to enable synchronously responsive updates of the projected screen in VR once the physical phone was touched. We also enabled the fingertip touch pointer (a white sloid dot) on the mirrored screen to indicate where the touch was registered, visually assisting the interaction.

Although the Full HD or even higher screen resolution of modern mobile phones provide enough pixels to fulfill the textured virtual screen theoretically, some content details, especially for text, may get lost, and a blurry or pixelated visual feeling could be observed in the VR view. This is because the copy of the mobile phone's screen can only be displayed in a small region inside the 3D virtual space at the limited VR display resolution.

3.4. Hand segmentation and visualization

To bring the 3D hand input into the VR scene, we first use an RGB-Depth sensor (Intel RealSense SR305[12]) to segment out the hand from the background with the depth frame Fig. 5a. This was done by removing the points located outside of the required depth range (less than 0.1 meters or more than 0.8 meters from the camera center). We then detected the hand region from the foreground by converting the foreground color image from the RGB to HSV color space and applying the skin-color probability detection

[10] https://www.ptc.com/en/products/augmented-reality/vuforia/

[11] https://github.com/Genymobile/scrcpy/

[12] https://www.intelrealsense.com/depth-camera-sr305/

[22] to get the skin-color pixels as the final hand region. With the intrinsic and extrinsic parameters of the color camera and infrared camera, we finally can visualize the hand region in a point-cloud format in the 3D virtual space (Fig. 5b) by uploading the aligned color and depth values to the GPU for real-time rendering.

3.5. Implementation and performance

We ran the VR system on a desktop computer (Intel Core i7-8700 CPU @3.2GHz, 32G RAM, GTX 1080 GPU, Windows 10 OS, Intel Wi-Fi 6 AX200 with Wi-Fi 6 support). The smartphone (Samsung Galaxy Note 10 Plus, 12G RAM, 6.8-inch Quad HD screen at a resolution of 3040×1440, Wi-Fi 6 enabled) projected the mobile screen to the VR computer via our private network connection (at a bandwidth of 530 Mbps with 170 ms delay on average).

The mobile phone screen refresh rate is 60 Hz natively, while the mirrored screen was transferred at 60 Hz as well since no explicit frame drop was found from the capturing rate of the Scrcpy plugin. However, during the final rendering step, the mobile screen was textured in the VR system at a resolution of 2476 × 1173, dropping to 48 frames per second (FPS) while still ensuring a clear and smooth presentation of the screen content to the VR user. The controller-based tracker provided stable tracking (4.36 ± 2.91 mm) [33] at an average frequency of 91 Hz, and the total weight added by the controller for the phone tracking was about 240g. The RealSense SR305 camera for hand detection ran at 60 FPS with a resolution of 640 × 480 and 69 by 54° FoV. The point-cloud rendering was refreshed at about 78 FPS on the Rift S display (Supporting up to 80 FPS), which took up 80% of the GPU capability.

4. Smartphone touch interaction in VR

We designed the first user study with the primary goal of evaluating the tangible AV interface that we developed for the mobile phone in a fully immersive virtual environment, and investigating the benefits and limitations of this interface in VR compared with the direct interaction with mobile phones in reality. In this case, our primary independent variable was the type of interfaces presented and used for the mobile interaction (*Direct touch in physical reality vs. Augmented Virtuality operation in VR*). The dependent variables were different perspectives of the user experience (easiness, workload, system usability, and so on). We tested everyday mobile tasks based on mobile screen touching with both interfaces to collect usability data and feedback for our evaluation.

4.1. Study environment and set-up

We set up our experiment in a small office room with a standard desk and chair. Although the user could walk around holding the mobile phone while wearing the VR headset, they were asked to sit on the chair and hold the mobile phone with two hands. This setup enables natural and flexible operations with the smartphone, where users can choose to use the phone in midair or put their elbows on the desk edge to obtain physical support, as shown in the Fig. 6. The experiment was conducted our experiment under a controlled lighting situation by keeping the room curtains closed. Although the color filter is compatible with a broad range of skin colors in different lighting conditions, the controlled environment makes robust and high-quality virtual hand capturing and rendering.

To eliminate any potential influence from the environment, we also created a virtual scene with a similar style and layout as the physical office. For example, a virtual desk is placed in front of the user in VR with the same color and size as the physical one, as partially shown in Fig. 5b. Besides, a controller is located at the corner of the desk to align the virtual desk with the physical one, so that users can naturally put their hands on the virtual desk and receive consistent visual and touch feedback.

Fig. 6. The experiment setup. The user sits on the chair and holds the mobile phone with two hands while placing their elbows on the desk for support.

4.2. User study I

In this study, we would like to verify if most of the essential everyday mobile tasks based on touchscreen interaction could be operated correctly and naturally in the VR environment. We selected five typical smartphone activities, and each was tested under both the real and virtual environments. For example, the user would first answer the phone or reply to a text message directly on the actual phone with the touch input without any VR gears. The participant would also complete the same tasks on the virtual phone with our AV interface by wearing the Oculus headset.

Our main research questions for this study were:

(Q1) How can we use a smartphone with screen-touch input in a VR environment with the same interaction methods as in reality without taking off the VR headset?

(Q2) What is the usability difference of the AV interface compared with physical input in reality for typical mobile phone applications?

Based on design experience and prior research [11–13], we made the following research hypotheses:

(H1) By tracking the smartphone's pose, mirroring its screen content, and capturing the user's 3D hand data, we could bring a fully-featured smartphone and its touch-based interaction into a virtual environment.

(H2) No significant difference would be found in common mobile interactions when using a smartphone in VR with our AV interface.

4.2.1. Experimental design

We used a within-subject design for the study that each participant would experience both conditions. The participants were asked to use the mobile phone as usual without the VR equipment, which was considered as our control condition (*Reality*). We then brought the mobile phone with the same screen-touch interaction into VR through the AV interfaces as the comparison (*AV*). The order of the interface conditions was counterbalanced between participants.

Each participant experienced five different applications, corresponding to five types of typical mobile phone activities under both conditions, as described below:

- *Video call* The participant makes a video call to the host (the author) via *Skype* and discusses some random topics for about a minute.
- *IM (Instant_messaging)* The participant uses *Facebook Messenger* to communicate with the host about their feelings by sending

and receiving at least five pieces of messages, including plain texts, gif pictures, or emojis.

- *SNS (Social_networking_service)* The participant checks new feeds from *Twitter* and posts their comments under one piece of interesting news.
- *Media Consumption* The participant watches a short video from *YouTube* for one minute and then tries to jump to the end using the playback interface.
- *Photography* The participant takes a photo of the local environment or makes a selfie, and then reviews the photo in *Google Photo* and shares it with the host.

Although people may use different mobile phone applications instead of the specific ones used in the study, the functions and user interfaces for each category are very common. Participants can quickly get used to the testing applications in our training sessions. Temporary accounts for these applications were created and logged in before the user study, so no personal data was recorded.

4.2.2. Experimental procedure

Before the study started, participants were introduced to the study design and the overall setup. The experiment began with the participants signing a consent form, answering demographic questions, and describing their VR/AR experience. The participants were then explained about the new system and the experiment tasks, followed by a training session in which participants tried each interface condition for all tasks (1 minute) to get familiar with the system. After the training session for each condition, the five tasks mentioned in Section 4.2.1 were tested in a randomized order for a longer time (2 minutes). Participants were asked to complete predefined questionnaires regarding the interface usability once all tasks were completed. There was a 1-minute short break between each task and a 10-minute long break applied between each condition. When all sessions (ten tasks in total for each participant) were finished, a short interview was conducted to collect more subjective comments. The whole procedure lasted about 60 minutes on average, and the participants could stop the experiment anytime during the study if needed.

Due to the Covid-19 pandemic, strict safety protection and sanitation rules were followed throughout the study. The headset and the mobile phone were carefully cleaned with wet sanitation tissues before each session. Participants' hands were also sanitized before and after the study.

4.2.3. Measurements

We used several subject questionnaires to evaluate the user experience across five testing tasks under each condition. We used the Single Ease Questionnaire (SEQ) [34] for measuring the total interface difficulty, the NASA Task Load Index Questionnaire (TLX) [34] for measuring the mental and physical workload, the System Usability Scale (SUS) for measuring the usability of the interface, and the User Experience Questionnaire (UEQ) [35] for measuring the general user experience. After completing both conditions, participants were asked the following interview questions; *1) What are the user experience differences between the two conditions on each task; 2) What are the pros and cons of the AV interface in the virtual environment?*

4.3. Result I

In this section, we report on the first study results with statistical methods regarding the usability of our developed AV interface for mobile interaction tasks compared with similar interaction in reality. The mean difference was marked significant at the 5% level (α=.05), and the threshold was adjusted with the Bonferroni correction for multiple post-hoc comparisons unless noted otherwise.

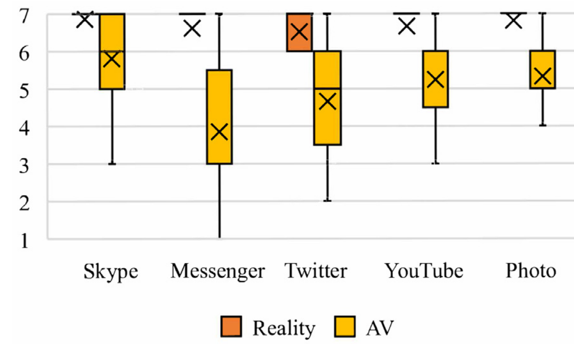


Fig. 7. SEQ results (Error bar: +/- *SE*; ×: Mean; The higher, the better).

Table 1
SEQ statistical analysis result.

	Skype	Messenger	Twitter	YouTube	Photo
Z	−3.125	−3.661	−3.539	−3.444	−3.361
p	.002	<.001	<.001	.001	.001

Table 2
NASA-TLX statistical analysis result.

	Skype	Messenger	Twitter	YouTube	Photo
Z	−7.380	−8.838	−7.511	−7.905	−7.333
p	<.000	<.000	<.000	<.000	<.000

We summarize qualitative feedback collected from the participants from the post-experiment interview.

We recruited 21 participants (6 females and 15 males) from the university campus, with their ages ranging from 22 to 45 years old (M=33.4, SD=5.9). Fifteen of them (71%) reported having VR experience, while six (29%) had never used any VR devices. We confirmed that all participants had their personal smartphones and were familiar with the standard touch-based interfaces. On average, their everyday smartphone usage time was 1-2 hours for 24%, 2-3 hours for 38%, and more than 3 hours for the rest (38%). Their primary usage for smartphones (multi-choice) was receiving and sending messages (81%), general web browsing (67%), photographing (52%), phone calls (38%), video/audio calls (33%), and social media checking (33%).

4.3.1. Task difficulties

The SEQ is a 7-point single-question Likert scale regarding the total difficulty of the task (1:Very Difficult, 7:Very Easy), and we collected responses from the five mobile interaction tasks with this scale. The Wilcoxon Signed-Rank test was conducted to compare the statistical difference between two conditions on each sub-task. Significant differences were found in all five categories between the two conditions, as shown in Fig. 7. The mobile interaction in reality was considered significantly easier than that in VR for the five tasks (Table 1). For example, we found that in the AV condition, participants felt that text input was pretty tricky (M=3.857, SE=0.386).

4.3.2. Workload

To evaluate the participants mental and physical efforts in each condition, we used the NASA-TLX questionnaire, which consists of six rating items within a 100-point range with 5-point steps (0:very low, 100:very high). The Wilcoxon Signed-Rank test showed a significant difference in workload between the two conditions across all five tasks (Table 2). Fig. 8 shows the average TLX rating results of each condition. Participants using the AV interface in the VR environment had a significantly higher workload in all

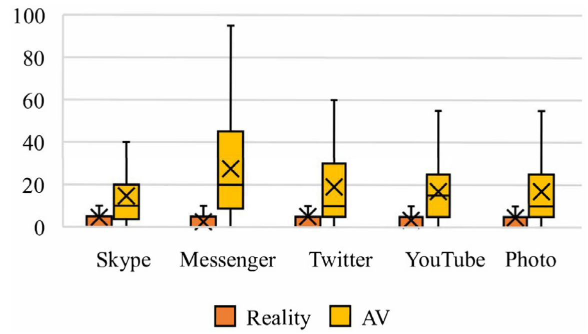


Fig. 8. NASA-TLX results (Error bar: +/- *SE*; ×: Mean; The lower, the better).

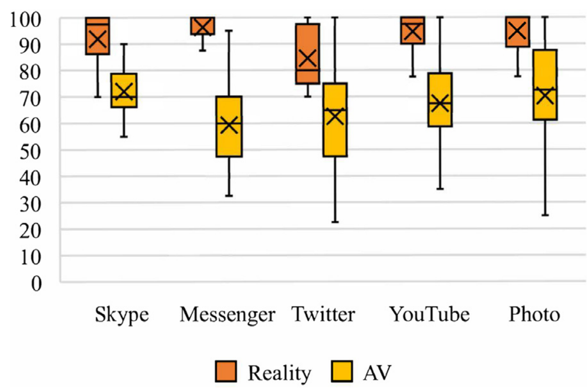


Fig. 9. SUS results (Error bar: +/- *SE*; ×: Mean; The higher, the better).

Table 3
SUS statistical analysis result.

	Skype	Messenger	Twitter	YouTube	Photo
Z	−3.922	−4.017	−3.827	−3.921	−3.726
p	<.000	<.000	<.000	<.000	<.000

tasks. We also noticed that messaging had a pretty high workload with the AV method, which aligned well with the SEQ results.

4.3.3. System usability

We used the SUS questionnaire to evaluate the system usability, consisting of 10 rating items with five response options (from Strongly Disagree to Strongly Agree). A SUS score of 68 or above is viewed as above average system usability. We reviewed the results of five tasks in the two conditions (Fig. 9), and the Wilcoxon Signed-Rank test showed significant differences in system usability between the two interfaces across all five tasks. As shown in Table 3, the system usability of the physical application interaction (control condition) was significantly higher than with the AV interface. However, the SUS results revealed that the usability of the calling (M=72.024, SE=1.994) and the photographing (M=70.238, SE=5.201) operations in the AV mode were both higher than the average score, while the augmented message typing had the lowest score (M=59.400, SE=3.380). In contrast, as all participants were very familiar with touch-based mobile interaction, the SUS scores from the five tasks in the real world were higher than 70.

4.3.4. User experience

The UEQ survey was used to evaluate the overall user experience of the system. The scales were adjusted from between 1(very bad) to 7 (excellent) to between -3 (very bad) to 3 (excellent). All the operations in both conditions notably obtained a positive mean value (Fig. 10). With the Wilcoxon Signed-Rank test, no significant difference was found on the calling task (Z=-1.226, p=.220) between the two conditions. However, there were significant differences (p<.001) in all other tasks.

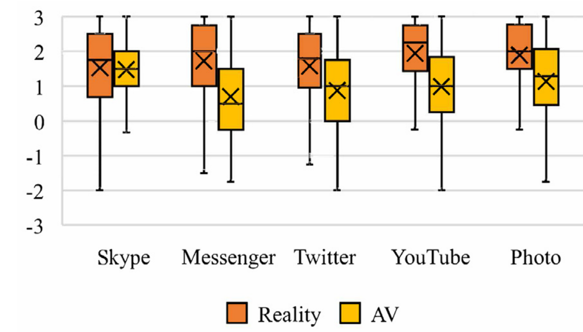


Fig. 10. UEQ results (Error bar: +/- *SE*; ×: Mean; The higher, the better).

5. Augmented smartphone typing in VR

Some people may have an occlusion issue in daily scenarios while typing on a small mobile screen keyboard as their fingertip might cover the screen target, creating many typing errors and a low typing speed. In the AV scene, we found that the typing experience was rated challenging and had the lowest usability score among the five tasks, as presented in Section 4.3.

The color and transparency of the 3D hand in the AV interface can be changed to improve the on-screen content's visibility and the typing experience on the virtual phone keyboard. We designed a follow-up study to explore if the augmented hand visualization style could improve text typing with the smartphone's on-screen keyboard. The study focused on evaluating the performance and experience of an augmented hand visualization regarding smartphone text typing. We chose skin color as the independent variable because colors can psychologically affect human visual search and attention [36].

5.1. User study II

Based on prior research [17,22,37], we chose to change the rendering color and transparency of the user's virtual hands for the augmented typing. The main research question for this study was:

(Q3) How do the color and transparency of point-cloud hands affect typing performance and user experience on smartphones in a VR environment?

Our research hypotheses were:

(H3) Using skin-colored hands for text input on the phone in VR helps with embodiment and the immersive experience compared to using other solid colors.

(H4) Adding transparency to virtual hands would increase the performance and reduce the occlusion issues during typing and improve the overall typing user experience.

5.1.1. Experimental design

We kept the exact system implementation and experiment setup described in Section 4.1 for the second study, except for the rendering configuration of the hand point-cloud. In specific, we added an extra configuration for users to choose the virtual hands' rendering color and transparency beyond the real skin texture. As shown in Fig. 11, the hands can be rendered in a realistic style or with a solid color, and the point-cloud could also be semi-transparent with a see-through attribute. The transparency value was determined in a pilot study concerning the balance of the virtual hands' penetrability and realism. Inspired by the hand repre-

(a) (b) (c)

Fig. 11. The different virtual hand visualization styles: (a) Realistic skin-colored non-transparent condition, (b) Augmented skin-colored semi-transparent condition, and (c) White-colored semi-transparent condition.

sentation style in Oculus Quest system[13], we selected the white color as one of our color conditions. It would allow the contour to be seen on top of the white-colored mobile keyboard to reduce the visual distraction from the noisy edge of the point-cloud data. In this study, three different visualization conditions were implemented for our comparison:

- *C1* Skin-colored non-transparent hands: The virtual hands were rendered with a real skin texture (Fig. 11a).
- *C2* Skin-colored semi-transparent hands: The skin-colored virtual hands were rendered with 30% opacity (Fig. 11b).
- *C3* White-colored semi-transparent hands: the virtual hands were rendered in a solid white color with 30% opacity (Fig. 11c).

5.1.2. Experimental task

With the same apparatus used in the first study, we developed two different typing tasks for users with three interface conditions. The first task was to type single characters, including small and capital letters (a-z, A-Z) and numbers (0-9). One of the characters was randomly presented to participants at a time. The character was renewed after typing on the keyboard, even if the typed character was incorrect. The default keyboard layout included all lowercase letters and numbers, while the *SHIFT* key needed to be used to input a capital character. The second task was word typing, and participants needed to type the whole word with a random length of 2 to 12 characters. The presented word would be updated after the space key was pressed. During the typing, the user could voluntarily choose to use the backspace key to correct typing errors.

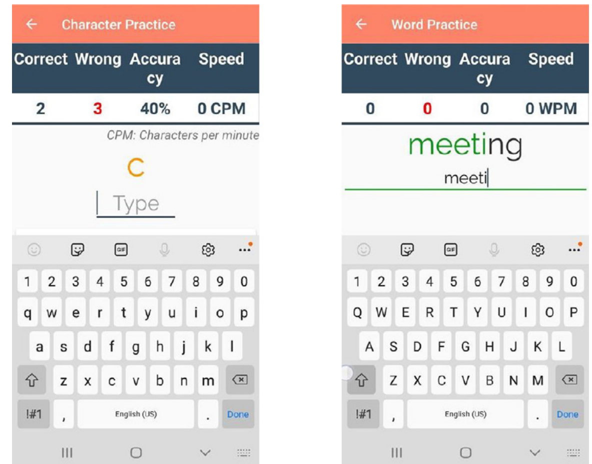
In each typing task, the participant had one minute to type on the mobile application[14] and the accuracy and the total number of typed characters or words were logged by the software. Meanwhile, the participants were asked to type as fast and accurately as possible. Screenshots of the sample test and results were shown in Fig. 12a and 12b.

5.1.3. Experimental procedure

We recruited the same participants from the first study in the same order. A brief introduction of upcoming tasks was given to help participants understand the basic concept. Participants were trained to get used to the AV typing with new hand rendering styles and learn how to use provided three styles with the mobile keyboard in the VR environment. They had 6 minutes to freely practice typing characters and words with all styled interfaces in the typing software. Participants had a 2-minute test (1 minute for characters and 1 minute for words) with each style condition, and all conditions were conducted in a counterbalanced order to avoid any learning effect. After each test, participants were asked to answer a few related questionnaires. Once all the tests were

[13] https://www.oculus.com/quest-2/

[14] https://play.google.com/store/apps/details?id=com.aswdc_typingspeed



(a) (b)

Fig. 12. Screenshots of the typing test software: (a) Character typing test; (b) Word typing test.

finished, participants were expected to rank the three conditions and provide subjective opinions about the interface and system via an interview. The typing accuracy and speed in each experimental condition were automatically logged. The same COVID-19 safety precautions were continued as well.

In the first user study, we enabled auto-correction for the message typing to reproduce a practical daily application case. However, in this test we disabled the auto-correction for the testing software for a basic result. The user study took about 30 min on average for each participant.

5.1.4. Measurements

We collected both objective and subjective data. For example, the typing speed and typing accuracy were collected in both character typing and word typing tasks. The typing speed is the number of characters or words participants typed per minute. The typing accuracy describes the proportion of the correct characters or words entered in the total number of entries. The two parameters can be obtained directly from the typing software's report. In addition to the questionnaires used in the AV condition of the first study, we also added the Slater-Usoh-Steed Questionnaire (SUSQ for short to distinguish from SUS) [35] to measure the immersive feelings in VR with different rendering styles.

5.2. Result II

In this section, we report on the experiment results with a statistical analysis regarding the augmented hand visualization method for AV smartphone typing.

5.2.1. Performance

We applied the Shapiro-Wilk test and found that some of the conditions were not following a normal distribution, so we used the Repeated-Measures ANOVA for our augmented typing accuracy and speed analysis. In terms of the typing accuracy, we found no significant difference in typing characters between three different interfaces ($F(2, 60)=.068$, $p=.935$), or typing words ($F(2, 60)=.452$, $p=.639$). In terms of the typing speed, we also found no significant difference in typing characters between the three different interfaces ($F(2, 60)=.064$, $p=.938$), or with typing words ($F(2, 60)=.250$, $p=.780$). However, as shown in Fig. 13, by comparing the accuracy of the AV interface such as C1 with the physical typing interaction, C1 provided a reasonably high accuracy (84% on character typing,

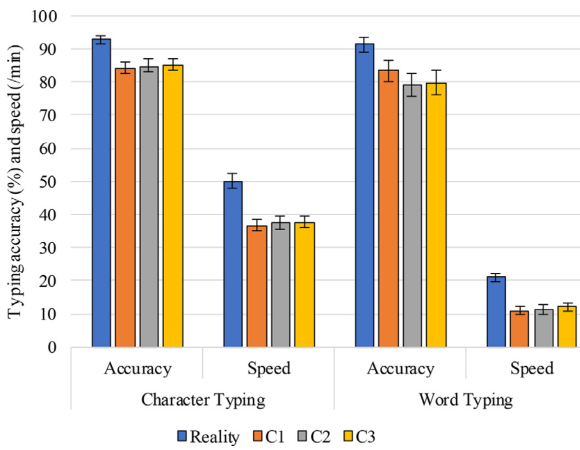


Fig. 13. Results of the typing accuracy (%) and speed (/min) in two typing tests (Error bar: +/- *SE*).

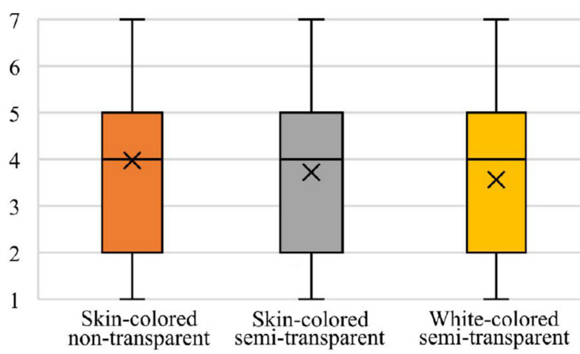


Fig. 14. SUSQ results (Error bar: +/- *SE*; ×: Mean; The higher, the better).

84% on word typing) that were not much less accurate (93% and 91% respectively) from the control condition.

5.2.2. Task difficulties and workload

We measured the ease of text typing in the VE with the three rendering methods. We used the Friedman Test and found that there was no statistically significant difference between all three conditions ($\chi^2(2)=0.226$, $p=.893$, C1 ($M=4.38$, $SE=0.387$), C2 ($M=4.33$, $SE=0.333$), C3 ($M=4.71$, $SE=0.391$)). We tested the workload of three AV interfaces and found no significant difference ($\chi^2(2)=0.199$, $p=.905$, C1 ($M=42.19$, $SE=3.25$), C2 ($M=42.19$, $SE=3.25$), C3 ($M=42.19$, $SE=3.25$)). So, in this case, the hand rendering difference with color or transparency did not influence the task difficulties or workload.

5.2.3. System usability and immersiveness

In terms of the SUS, all three hand rendering methods were rated around the average level of usability (C1: $M=62.98$, $SE=4.35$; C2: $M=65.83$, $SE=3.78$; C3: $M=61.79$, $SE=5.23$). While C2 received the highest ratings, the difference between the conditions was not statistically significant based on a Friedman Test ($\chi^2(2)=1.351$, $p=.509$).

To measure the immersiveness inside the VE with the AV interface, we recorded the SUSQ and found a significant difference between three AV rendering interfaces using the Friedman Test (($\chi^2(2)=9.980$, $p=.007$). Followed by the Wilcoxon Signed-Ranks Test, the skin-colored non-transparent rendering method was rated significantly more immersive than the white-colored semi-transparent style ($Z=-3.338$, $p=.001$, C1 ($M=3.98$, $SE=0.159$), C2 ($M=3.71$, $SE=0.144$), C3 ($M=3.56$, $SE=0.141$)). Fig. 14 shows the mean immersion score from the SUSQ questionnaire for three rendering interfaces.

5.2.4. User experience

This questionnaire's results were also adjusted between -3 (very bad) to 3 (excellent), and the results indicated that the experience was above the average score and positive. The Friedman Test showed a significant difference ($\chi^2(2)=6.858$, $p=.032$) among the three rendering styles. With post-hoc analysis using the Wilcoxon Signed-Ranks Test, the natural hand color was found to be significantly better than the semi-transparent skin-color ($Z=-3.478$, $p=.001$) and the semi-transparent white-color ($Z=-2.689$, $p=.007$). However, there was no significant difference between the natural hand color condition and the semi-transparent skin-color condition ($Z=-0.320$, $p=.749$).

6. Discussion

In this section, we discuss the results of our studied research questions in more detail with experimental observations and post-experiment interviews.

6.1. Bringing full-featured smartphone into VR

Our developed AV system visually and tangibly brought full smartphone functions and features into the VE while keeping the same interaction modalities as in reality. Using the customized tracker, the virtual phone model was well aligned with the actual phone, so that the user could feel the smartphone's shape through hands. The hands were also captured and rendered with a dense point-cloud in the virtual world. Both the dynamic visual and tactile feedback provided users with a consistent multimodal experience coherently with the experience of using an actual smartphone even in VR [38].

The five tasks in the first study (Section 4.2) tested the smartphone's functions and its usability of fundamental interactions in VR. With the provided system, participants could complete all smartphone operations in VR while wearing the headset. As expected, the easiness and overall workload were rated based on the task complexity. The video call received the lowest rating while the message input got the highest rating [39]. Regarding the system usability, the calling and photographing tasks got the highest score while the message typing was still rated as the lowest. For other tasks (e.g., browsing Twitter or watching YouTube videos), most interactions were handled by swiping the screen or clicking larger-size buttons or icons, rated higher in most usability questionnaires than the typing task.

From the post-experiment interview regarding the AV interface, some participants said that *"it gave a seamless experience with the real phone while still being immersed in the virtual world"* (User 1), *"with no breaking the VR experience"* (User 9), and created *" good link to reality"* (User 20). User 11 mentioned that *"the system is surprisingly good to play with the real phone"*. User 20 also commented that the system *"might motivate to bring people into VR world"*. On the other hand, there are also some negatives to be noticed. User 19 mentioned that *"random misalignment errors and the rendered visual effects with noise worsen the button click, especially when some of the buttons were located very close with each other"*. User 15 claimed that he/she *"felt fatigued after some time"*. However, they commented that overall the system offered no significant difference with the experience of using a smartphone in reality. Therefore, H1 (bringing the full-featured mobile phone into VR) is confirmed.

6.2. AV vs. reality smartphone interaction

Examining our research hypotheses from the results in Section 4.3, we found that H2 (no difference between Reality and AV interaction) was rejected as the usability of the AV interface for mobile applications in the VR environment was significantly lower than the direct touch-based mobile interaction.

Although with current VR hardware, it might not be easy to use for complicated tasks that require low-level details, like typing and reading messages. Our AV mobile interface was still considered good for certain simple tasks like answering a phone call, checking videos, or taking photos in VR. A higher resolution VR display may be required to bring the mobile screen with its content into the VR environment for accurate and comfortable interaction.

The AV typing was not as accurate or fast as the direct physical touchscreen typing. However, the novel method provided a fairly accurate input solution with a reasonable speed for users while wearing the VR headset compared to the default controller-based typing for mainstream VR systems. This means that more text-based social media applications like Twitter on the smartphone could benefit from the AV interface for VR scenarios potentially.

6.3. Improving phone text typing in VR

As discussed in the first study, we found that AV typing needed quite a lot of mental and physical effort to focus on, and eyes could get tired after some time (User 1). Some reading/typing tasks needed extra self-interpretation based on what was shown (User 19). In addition to the clarity issue caused by the VR display resolution, some participants also noted that the typing difficulty was also induced by 1) some delays happening when typing too fast (User 21), and 2) the movement mismatched the virtual and real button by heavy clicking. For example, one user pressed "W" on the physical keyboard when intending to type Q in the virtual view (User 19). This problem could be addressed with more accurate and responsive real-virtual object tracking in the future.

Examining our research hypotheses in Section 5.2, we found that H3 (skin color helps embodiment and immersiveness) was confirmed as the skin-colored style was rated significantly higher with a better user experience than the solid color one, no matter whether the transparency was enabled or not. Our result was aligned with previous research. For example, the color of avatar hands could influence human psychological perceptions in VR [40], and the skin-colored presentation could help users perceive higher levels of virtual presence and enhance the VR experience [17,22].

H4 (transparency reduces the occlusion issue) was rejected. Although the white transparent rendering method was rated significantly lower regarding the immersiveness of the virtual environment, both transparent interfaces had no performance difference from the non-transparent condition. Moreover, the random edge noise of the point-cloud hand made it a bit difficult to concentrate (User 20) during the test. The result was different from prior research that a transparent hand could enhance environment perception [26] and typing performance with a keyboard [17]. One possible reason would be that the size of the phone keyboard is much smaller than that of a typical desktop keyboard. There could be two or three virtual keys for the mobile phone within the region of a fingertip. Users could not able to identify which key was actually hit below the relatively larger fingertip, even with the transparency clue. The dense distribution of the virtual keyboard layout might also reduce the transparency advantage.

6.4. Psychological effect

Instead of the mainly focused performance and usability results, we also discovered a few psychological effects for our AV interaction. First of all, User 20 mentioned that he/she felt insecure because he/she could not know if someone was around to check his/her mobile screen that might have some private content that he/she did not prefer to share with others. Although the user can manipulate the mobile phone through the AV interface, he/she still needs to understand the people around to get the context of the surroundings. The AV technology might make people un-social at some point since people do not need to break the VR experience to complete most mobile actions and keep themselves inside the VR for a longer time, a concern mentioned by User 14. User 14 also mentioned the Uncanny Valley theory [41]. Specifically, the AV interaction could make the VR environment too good, which would not always be a great idea. For example, he/she claimed that "*In the beginning, this system feels very cool, but later I felt a bit scared, because the VR world is so real, which mixes me with the reality in a bad sense*" [sic]. Extra features need to be considered from the psychological perspective to reach a well-balanced point by mixing the real and virtual content to a certain extent.

6.5. Limitations

Our AV interface had its limitations that could be improved in the future. A few users complained that the VR headset itself was too heavy to wear with all of the sensors attached and was not very friendly to people with glasses (User 10, User 11). Although we carefully considered the weight distribution of the attached tracker on the smartphone to provide a better balance, the current kit was still heavy to hold for long-term usage. A customized phone case with embedded photodiodes for tracking could be developed to replace the current solution, and provide a lightweight VR tracking result with more reliable alignment between the screen content and the virtual phone model. Since we used a visual-based skin-color detection algorithm, the hand segmentation could be significantly affected by the environmental lighting condition. In some cases, skin-color like content, such as pictures on the mobile screen, might not be removed appropriately from the hand region.

The user study in this research focused on the seated pose for participants because of the above limitations. In contrast, interacting with the smartphone while moving around for participants could be investigated once self-contained VR HMDs could be deployed for our implementation.

7. Conclusion and future work

In sum, we present an AV interface that enables a user to operate a real smartphone in a VR environment. The system allows the user with a VR headset to hold a mobile phone and operate the touchscreen with their real hands, successfully bringing complete mobile interaction into VR in real time. Two user studies were conducted to evaluate the advantages and limitations of this novel interface based on typical mobile touch-based applications in the VR environment, especially for text typing on the screen. The results showed that our system successfully connected the actual smartphone with the virtual world, and most users felt the same experience in VR as using their smartphones in reality. However, participants reported that our AV interface was significantly different from the familiar mobile interface in the real world, with less usability caused by the hardware and software limitations at this moment. They commented that the skin color textured rendering interface could offer a better user experience and immersiveness than other rendering styles. We provided a prototype and its method for researchers and developers to evaluate the daily mobile phone interaction in VR and explore possible design implications before the mature VR phone solutions became widely available.

In the future, we plan to explore the design and development of novel robust manipulation methods for virtual objects in VR environments based on augmented mobile screen-touch input that we discussed here. We would also like to explore asymmetric VR interaction, which will combine a classic VR controller or free-hand gestures for 3D rough spatial operation with mobile touch input for 2D precise adjustment. This work focused more on bringing the mobile phone into the VR scene with old interaction patterns rather than demonstrating new phone-based interaction paradigms for VR. Such work can be conducted in the future to improve the usability of the VR phone and enrich phone-based interactions in virtual worlds.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Mark Billinghurst, one of the authors listed on this paper, also serves as a guest editor for the Computer & Graphics journal.

CRediT authorship contribution statement

Huidong Bai: Conceptualization, Investigation, Methodology, Project administration, Software, Formal analysis, Writing - original draft. **Li Zhang:** Investigation, Data curation, Formal analysis, Software, Writing - original draft. **Jing Yang:** Formal analysis, Writing - original draft. **Mark Billinghurst:** Supervision, Funding acquisition, Resources, Writing - review & editing.

Acknowledgments

We would like to thank the Tertiary Education Commission/Te Amorangi Mtauranga Matua, New Zealand for funding this research through Grant 9077/3716890.

Supplementary material

Supplementary material associated with this article can be found, in the online version, at doi:10.1016/j.cag.2021.04.004.

References

[1] Takashina T, Ikeya M, Tamura T, Nakazumi M, Nomura T, Kokumai Y. Real-virtual bridge: operating real smartphones from the virtual world. In: Proceedings of the 2018 ACM international conference on interactive surfaces and spaces. In: ISS 18. New York, NY, USA: Association for Computing Machinery; 2018. p. 449–52. doi:10.1145/3279778.3279921.

[2] Milgram P, Takemura H, Utsumi A, Kishino F. Augmented reality: a class of displays on the reality-virtuality continuum. In: Telemanipulator and telepresence technologies. In: CHI '20. International Society for Optics and Photonics; 1995. p. 282–92. doi:10.1117/12.197321.

[3] Hartmann J, Vogel D. An evaluation of mobile phone pointing in spatial augmented reality. In: Proceedings of the extended abstracts of the 2018 CHI conference on human factors in computing systems. In: CHI EA 18. New York, NY, USA: Association for Computing Machinery; 2018. p. 1–6. doi:10.1145/3170427.3188535.

[4] Henrysson A, Billinghurst M, Ollila M. Virtual object manipulation using a mobile phone. In: Proceedings of the 2005 international conference on augmented tele-existence. In: ICAT 05. New York, NY, USA: Association for Computing Machinery; 2005. p. 164–71. doi:10.1145/1152399.1152430.

[5] Larsen BD, Bærentzen JA, Christensen NJ. Using cellular phones to interact with virtual environments. In: Proceedings of the ACM SIGGRAPH 2002 conference abstracts and applications. In: SIGGRAPH 02. New York, NY, USA: Association for Computing Machinery; 2002. p. 274. doi:10.1145/1242073.1242287.

[6] Liang H-N, Shi Y, Lu F, Yang J, Papangelis K. Vrmcontroller: an input device for navigation activities in virtual reality environments. In: Proceedings of the 15th ACM SIGGRAPH conference on virtual-reality continuum and its applications in industry - Volume 1. In: VRCAI 16. New York, NY, USA: Association for Computing Machinery; 2016. p. 455–60. doi:10.1145/3013971.3014005.

[7] Steed A, Julier S. Design and implementation of an immersive virtual reality system based on a smartphone platform. In: Proceedings of the IEEE symposium on 3D user interfaces (3DUI); 2013. p. 43–6. doi:10.1109/3DUI.2013.6550195.

[8] Choi I, Ofek E, Benko H, Sinclair M, Holz C. Claw: a multifunctional handheld haptic controller for grasping, touching, and triggering in virtual reality. In: Proceedings of the 2018 CHI conference on human factors in computing systems. In: CHI 18. New York, NY, USA: Association for Computing Machinery; 2018. p. 1–13. doi:10.1145/3173574.3174228.

[9] Mine M, Yoganandan A, Coffey D. Making vr work: building a real-world immersive modeling application in the virtual world. In: Proceedings of the 2nd ACM symposium on spatial user interaction. In: SUI 14. New York, NY, USA: Association for Computing Machinery; 2014. p. 80–9. doi:10.1145/2659766.2659780.

[10] Vinayak, Ramanujan D, Piya C, Ramani K. Mobisweep: exploring spatial design ideation using a smartphone as a hand-held reference plane. In: Proceedings of the TEI 16: tenth international conference on tangible, embedded, and embodied interaction. In: TEI 16. New York, NY, USA: Association for Computing Machinery; 2016. p. 12–20. doi:10.1145/2839462.2839490.

[11] Alaee G, Deasi AP, Pena-Castillo L, Brown E, Meruvia-Pastor O. A user study on augmented virtuality using depth sensing cameras for near-range awareness in immersive VR. In: Proceedings of the IEEE VRs 4th workshop on everyday virtual reality (WEVR 2018), 10; 2018.

[12] Chang Y-K, Huang J-W, Chen C-H, Chen C-W, Peng J-W, Hu M-C, Yao C-Y, Chu H-K. A lightweight and efficient system for tracking handheld objects in virtual reality. In: Proceedings of the 24th ACM symposium on virtual reality software and technology. In: VRST 18. New York, NY, USA: Association for Computing Machinery; 2018.

[13] Desai AP, PeŞa-Castillo L, Meruvia-Pastor O. A window to your smartphone: Exploring interaction and communication in immersive VR with augmented virtuality. In: Proceedings of the 14th conference on computer and robot vision (CRV); 2017. p. 217–24.

[14] Witmer BG, Singer MJ. Measuring presence in virtual environments: a presence questionnaire. Presence 1998;7(3):225–40.

[15] Bai H, Sasikumar P, Yang J, Billinghurst M. A user study on mixed reality remote collaboration with eye gaze and hand gesture sharing. In: Proceedings of the 2020 CHI conference on human factors in computing systems. In: CHI '20. New York, NY, USA: Association for Computing Machinery; 2020. p. 1–13. doi:10.1145/3313831.3376550.

[16] Han S, Liu B, Wang R, Ye Y, Twigg CD, Kin K. Online optical marker-based hand tracking with deep labels. ACM Trans Graph 2018;37(4). doi:10.1145/3197517.3201399.

[17] Knierim P, Schwind V, Feit AM, Nieuwenhuizen F, Henze N. Physical keyboards in virtual reality: Analysis of typing performance and effects of avatar hands. In: Proceedings of the 2018 CHI conference on human factors in computing systems. In: CHI 18. New York, NY, USA: Association for Computing Machinery; 2018. p. 1–9. doi:10.1145/3173574.3173919.

[18] Ge L, Cai Y, Weng J, Yuan J. Hand pointnet: 3d hand pose estimation using point sets. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition; 2018. p. 8417–26.

[19] Billinghurst M, Piumsomboon T, Bai H. Hands in space: Gesture interaction with augmented-reality interfaces. IEEE Comput Graph Appl 2014;34(1):77–80.

[20] Gao L, Bai H, Lindeman R, Billinghurst M. Static local environment capturing and sharing for mr remote collaboration. In: Proceedings of the SIGGRAPH Asia 2017 mobile graphics & interactive applications. New York, NY, USA: Association for Computing Machinery; 2017. SA 17

[21] Grubert J, Witzani L, Ofek E, Pahud M, Kranz M, Kristensson PO. Effects of hand representations for typing in virtual reality. In: Proceedings of the IEEE conference on virtual reality and 3D user interfaces (VR); 2018. p. 151–8.

[22] Schwind V, Knierim P, Tasci C, Franczak P, Haas N, Henze N. æthese are not my hands!g: effect of gender on the perception of avatar hands in virtual reality. In: Proceedings of the 2017 CHI conference on human factors in computing systems. In: CHI 17. New York, NY, USA: Association for Computing Machinery; 2017. p. 1577–82. doi:10.1145/3025453.3025602.

[23] Nahon D, Subileau G, Capel B. ænever blind VRg enhancing the virtual reality headset experience with augmented virtuality. In: Proceedings of the IEEE virtual reality (VR); 2015. p. 347–8.

[24] Tecchia F, Avveduto G, Carrozzino M, Brondi R, Bergamasco M, Alem L. [poster] interacting with your own hands in a fully immersive mr system. In: Proceedings of the IEEE international symposium on mixed and augmented reality (ISMAR); 2014. p. 313–14.

[25] McGill M, Boland D, Murray-Smith R, Brewster S. A dose of reality: overcoming usability challenges in vr head-mounted displays. In: Proceedings of the 33rd annual ACM conference on human factors in computing systems. In: CHI 15. New York, NY, USA: Association for Computing Machinery; 2015. p. 2143–52. doi:10.1145/2702123.2702382.

[26] Buchmann V, Nilsen T, Billinghurst M. Interaction with partially transparent hands and objects. In: Proceedings of the sixth Australasian conference on user interface - Volume 40. In: AUIC 05. AUS: Australian Computer Society, Inc.; 2005. p. 17–20.

[27] Bowman DA, Rhoton CJ, Pinho MS. Text input techniques for immersive virtual environments: An empirical comparison. In: Proceedings of the human factors and ergonomics society annual meeting, 46. SAGE Publications Sage CA: Los Angeles, CA; 2002. p. 2154–8.

[28] Yu D, Fan K, Zhang H, Monteiro D, Xu W, Liang H. Pizzatext: text entry for virtual reality systems using dual thumbsticks. IEEE Trans Vis Comput Graph 2018;24(11):2927–35.
[29] Boustila S, GuĘgan T, Takashima K, Kitamura Y. Text typing in VR using smartphones touchscreen and HMD. In: Proceedings of the IEEE conference on virtual reality and 3D user interfaces (VR); 2019. p. 860–1.
[30] Kim YR, Kim GJ. Hovr-type: smartphone as a typing interface in VR using hovering. In: Proceedings of the IEEE international conference on consumer electronics (ICCE); 2017. p. 200–3.
[31] Feit AM, Weir D, Oulasvirta A. How we type: Movement strategies and performance in everyday typing. In: Proceedings of the 2016 CHI conference on human factors in computing systems. New York, NY, USA: Association for Computing Machinery; 2016. p. 4262–73. doi:10.1145/2858036.2858233. CHI -16
[32] Lin J-W, Han P-H, Lee J-Y, Chen Y-S, Chang T-W, Chen K-W, Hung Y-P. Visualizing the keyboard in virtual reality for enhancing immersive experience. In: Proceedings of the ACM SIGGRAPH 2017 posters. In: SIGGRAPH 17. New York, NY, USA: Association for Computing Machinery; 2017.
[33] Jost TA, Nelson B, Rylander J. Quantitative analysis of the oculus rift s in controlled movement. Disabil Rehabil Assist Technol 2019;0(0):1–5. doi:10.1080/17483107.2019.1688398. PMID: 31726896
[34] Stanton N, Salmon PM, Rafferty LA. Human factors methods: a practical guide for engineering and design. Ashgate Publishing, Ltd; 2013.
[35] Laugwitz B, Held T, Schrepp M. Construction and evaluation of a user experience questionnaire. In: Proceedings of the symposium of the Austrian HCI and usability engineering group. Springer; 2008. p. 63–76.
[36] Green BF, Anderson LK. Color coding in a visual search task.. J Exp Psychol 1956;51(1):19.
[37] Farmer H, Tajadura-JimĘnez A, Tsakiris M. Beyond the colour of my skin: How skin colour affects the sense of body-ownership. Conscious Cognit 2012;21(3):1242–56. doi:10.1016/j.concog.2012.04.011.
[38] Allen PK. Integrating vision and touch for object recognition tasks. Int J Robot Res 1988;7(6):15–33.
[39] MacKenzie IS, Soukoreff RW. Text entry for mobile computing: models and methods, theory and practice. Hum Comput Interact 2002;17(2-3):147–98.
[40] Martini M, Pérez Marcos D, Sanchez-Vives MV. What color is my arm? Changes in skin color of an embodied virtual arm modulates pain threshold. Front Hum Neurosci 2013;7:438.
[41] Ho C-C, MacDorman KF. Revisiting the uncanny valley theory: developing and validating an alternative to the godspeed indices. Comput Hum Behav 2010;26(6):1508–18.