HandPainter – 3D sketching in VR with hand-based physical proxy

HandPainter - 3D Sketching in VR with Hand-based Physical Proxy

Ying Jiang
The University Hong Kong
yingjiang@connect.hku.hk

Congyi Zhang*
The University of Hong Kong
cyzh@hku.hk

Hongbo Fu
City University of Hong Kong
hongbofu@cityu.edu.hk

Alberto Cannavò
Politecnico di Torino
alberto.cannavo@polito.it

Fabrizio Lamberti
Politecnico di Torino
fabrizio.lamberti@polito.it

Henry Y K Lau
The University of Hong Kong
hyklau@hku.hk

Wenping Wang
Texas A&M University
The University of Hong Kong
wenping@cs.hku.hk

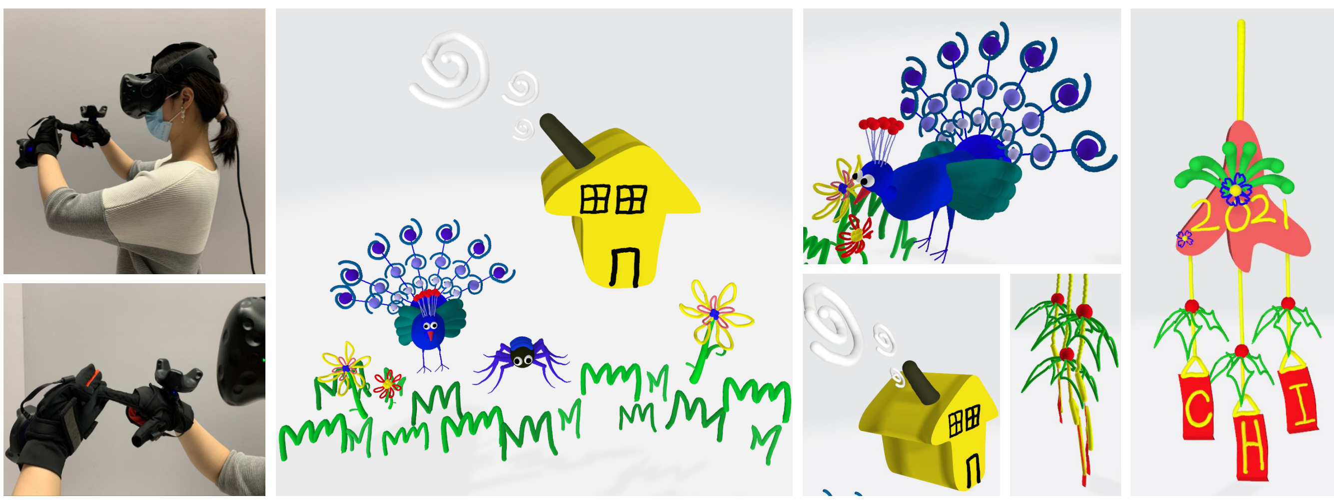


Figure 1: HandPainter is designed for novice users to create 3D concept designs in VR environments in a hand-based sketching manner.

ABSTRACT

3D sketching in virtual reality (VR) enables users to create 3D virtual objects intuitively and immersively. However, previous studies showed that mid-air drawing may lead to inaccurate sketches. To address this issue, we propose to use one hand as a canvas proxy and the index finger of the other hand as a 3D pen. To this end, we first perform a formative study to compare two-handed interaction with tablet-pen interaction for VR sketching. Based on the findings of this study, we design HandPainter, a VR sketching system which focuses on the direct use of two hands for 3D sketching without requesting any tablet, pen, or VR controller. Our implementation is based on a pair of VR gloves, which provide hand tracking and gesture capture. We devise a set of intuitive gestures to control various functionalities required during 3D sketching, such as canvas panning and drawing positioning. We show the effectiveness of HandPainter by presenting a number of sketching results and discussing the outcomes of a user study-based comparison with mid-air drawing and tablet-based sketching tools.

*Corresponding author

Yokohama '21, May 8–13, 2021, Yokohama, Japan

ACM ISBN 978-1-4503-XXXX-X/21/06…$15.00
https://doi.org/10.1145/1122445.1122456

CCS CONCEPTS

• **Human-centered computing** → **Virtual reality**; **User interface design**.

KEYWORDS

VR, 3D sketching, hand-based interaction

ACM Reference Format:

Ying Jiang, Congyi Zhang, Hongbo Fu, Alberto Cannavò, Fabrizio Lamberti, Henry Y K Lau, and Wenping Wang. 2021. HandPainter - 3D Sketching in VR with Hand-based Physical Proxy. In *Yokohama '21: ACM CHI, May 8–13, 2021, Yokohama, Japan.* ACM, New York, NY, USA, 13 pages. https://doi.org/10.1145/1122445.1122456

1 INTRODUCTION

Sketching is an effective tool to explore, evaluate, revise, and communicate ideas rapidly by freehand drawing [11]. Thanks to the great development in consumer-grade augmented reality (AR) and virtual reality (VR) hardware and software, it is possible to sketch 3D curves directly in mid-air [2, 10, 28, 44, 50]. More and more users exploit commercial VR painting tools (e.g., Tilt Brush, Quill, Gravity Sketch, and A-Painter) to sketch in mid-air freely and intuitively [6]. Nonetheless, precise mid-air sketching is challenging, especially for users with low spatial ability [5, 33], mainly because of the lack of a physical support [3].

In recent VR sketching systems [2, 13], tablets have been adopted as physical proxies to improve the drawing accuracy. Holding a tablet, however, would easily cause fatigue and discomfort [15, 39]. Hence, how to improve the drawing accuracy in 3D sketching without introducing additional hand-held hardware is a valuable problem to be addressed.

Regarding the use of physical proxies, Drey et al. [13] carried out a user study on a drawing system letting their study participants switch between 2D surface-based sketching (on a tablet) and 3D mid-air sketching. Their results showed that, because of the overhead associated with the switching, the users preferred to operate with 3D mid-air sketching alone. Therefore, investigating how to combine the 2D and 3D sketching modes by offering a seamless way to switch between them represents another interesting topic to explore [13].

As for hand interaction, more and more researchers turned their attention to methods based on bare hands, which provide the most familiar approach for manipulating the physical world [29] and, thus show a great potential for interacting with the virtual world as well. For example, considering general interaction tasks in VR, hand-based methods are regarded as easy-to-use and intuitive by users [40]. Focusing specifically on sketching tasks, hands can naturally provide physical proxies and could potentially replace tablets for proxy-based sketching. Hand-based physical proxies, however, have not been explored for VR sketching yet.

To better understand the unique advantages of hand-based proxies compared with tablet-based proxies, we conducted a formative study on two-handed interaction for VR sketching. The results of the formative study not only confirm the feasibility of hand-based proxies for VR sketching, but also show the possibility to use hand gestures for achieving a seamless switch between 2D sketching on a hand and 3D mid-air sketching. Based on the findings of the formative study, we devise a system, named HandPainter, which allows users to sketch both in mid-air and on a hand-based physical proxy intuitively without holding any device. As for 2D sketching, users can form a planar canvas with the non-dominant hand and use the index finger of the dominant hand to draw on the canvas. For switching to mid-air sketching, users simply use a pinch gesture with the dominant hand. Besides 2D proxy-based sketching and 3D mid-air sketching, our system also supports 3D deformation of planar strokes via a hand-based NURBS proxy and the creation of 3D sweep surfaces. Motivated by the design dimensions identified in the formative study, we also implement rescaling, color and width setting, moving, deleting, undoing, and duplicating operations to make it easy for the users to edit and assemble various components and to build complex objects conveniently. Instead of using costly motion capture systems adopted in previous tablet-based 3D sketching implementations [2, 13], we devise a more accessible hardware solution by building our drawing interaction kit on Manus VR gloves and HTC Vive trackers. Our comprehensive study with novice users and quantitative comparisons show the superiority of our system over VR-controller-based and tablet-based sketching methods.

In summary, the main contributions of this work encompass the following elements:

- we performed a formative study to compare two-handed interaction and tablet-pen interaction for VR sketching;
- we introduced a new idea of hand-based proxy for VR sketching, and developed the first prototype implementation of such a system, named HandPainter (Figure 1), which seamlessly integrates precise hand-based sketching and freehand mid-air sketching without relying on additional hand-held devices;
- we conducted a quantitative evaluation to assess the key components of HandPainter, and also showed the expressiveness of our system by showcasing a variety of 3D designs created by novice users.

2 RELATED WORK

Below we discuss the closely related work, including hand tracking techniques, hand-based interaction, and 3D sketching in AR and VR.

Hand tracking. Research works in the field of VR often make use of hand-held controllers or haptic devices [46, 54]. For instance, Transcalibur [45] is a reconfigurable hand-held VR controller tracked by one HTC Vive tracker and the associated Lighthouse tracking system, which was exploited to bring 2D haptic shape illusion in a VR environment. We take advantage of the same tracking system to get the global coordinates of users' hands. However, this controller needs to be hand-held. Rather than leveraging extra hand-held devices, our work focuses on exploring free, bare-handed interaction. Intuitive hand interaction is impossible without accurate hand tracking [31]. Optical tracking with RGB or RBGD cameras [18, 29, 42] achieved great results in tracking a single hand or two isolated hands. However, tracking both hands simultaneously with complex interactions using optical devices is very challenging, since optical systems are sensitive to occlusions [35] and self-occlusions. Exploiting wearable devices, such as data gloves, is the most accurate approach for tracking hand motion [9, 17, 31]. Since hand tracking is not the focus of our work, we simply adopt digital VR gloves for accurate hand tracking.

Hand-based interaction. It is convenient for humans to manipulate objects and communicate by gestures owing to the great maneuverability and expressiveness of hands [31]. The progress of hand-tracking technologies [35, 36] has contributed to the development of 3D authoring tools supporting hand interaction as an input means [30]. For instance, using a 3D printed robotic arm the RoMA system [38] allowed users to create well-proportioned tangible artifacts or to extend existing objects with the help of hand interaction combined with the 3D printing technology. Rather than using such technology, our work focuses on exploiting bare-handed interaction to fulfill both 3D freehand sketching [14] and 2D proxy-based sketching. Vinayak et al. [43] used hand motion to edit 3D shapes inferred from the geometry of the contact regions between a manipulated object and the hands themselves, allowing users to achieve desired shape deformations without learning or remembering any gesture. Inspired by [43], our system exploits gesture-free interaction for editing B-rep surfaces. It is worth observing that, with the growing popularity of tablets and smartphones, hand interaction has become a common way for letting users directly and intuitively manipulate items on a touch-screen without remembering input commands [26]; in a virtual environment, hand interaction plays a key role in improving object grasping and manipulation [48], as well as in enabling locomotion [34]. Despite that, hand interaction has often been investigated for simple tasks [20, 48], like pressing a button or making a selection. Instead, our work focuses on exploiting the expressiveness and flexibility of hand interaction to facilitate a complex task of 3D sketching in VR.

3D sketching in AR and VR. Great advancements in AR and VR propel the development of ever more sophisticated 3D sketching systems. Early 3D drawing systems adopted selection and manipulation primitives to create virtual objects [10], exploited surface drawing based on hand motion [44], weaved curves generated by a pen into an existing curve network [49, 50] displayed in stereoscopic glasses or CAVE-like systems [28], etc. Various VR applications inspired a new creative mode for painting, designing, and modeling. In VR, however, mid-air drawing accuracy decreased by 148% compared to traditional 2D drawing [3], implying that accurate spatial sketching was much more challenging. In both the Lift-off 3D sketching system [24] and the WireDraw system [53], visual clues were used to alleviate the problem due to the lack of control in 3D drawing. Keefe et al. [27] used a haptic-aided input technique to support the drawing of controlled 3D curves through space. Ye et al. [52] studied 3D absolute drawing errors under different input points and grip postures of a smartphone in mobile AR, and presented an interactive interface for fixing such errors. However, these approaches limited the freedom of the used input devices and that of hand-based interaction.

A common approach adopted by previous studies to cope with the inaccuracy of 3D sketching, which is mainly caused by the lack of support surfaces [3], is to exploit a physical tablet with a pen or an index finger as an input device for tablet-based sketching. For instance, Arora et al. [2] proposed SymbiosisSketch, which combines sketching in air and on a digital tablet to create detailed 3D designs. Mobi3DSketch [32] and VRSketchin [13] adopted similar approaches, using a finger to draw in mobile AR and a tracked pen to draw on a tablet in VR, respectively. However, holding a tablet might easily cause fatigue, since users need to keep the tablet stable with one hand while sketching with the other hand [15]. According to [39], the comfortable holding time of a tablet is only around 27 minutes. All these tablet-based sketching systems rely on a digital tablet to record the position of a pen in the tablet's local coordinate system, which is gathered using some screen-integrated sensors. Thus, a drawing set made up of a digital tablet and its pen could not be easily replaced by a more lightweight physical surface or a specially designed pen, e.g., a cardboard or a Logitech VR Ink pen. Furthermore, when using a tablet and a pen for 2D sketching in VR, the pen needs to be properly designed to keep the weight limited, and is often tracked by expensive systems. For instance, in [13], to precisely track the 3D position of the tablet and the pen, an expensive, professional tracking system by OptiTrack was used. Moreover, tablet-based sketching and 3D mid-air sketching were used separately because of the overhead associated with switching between the two modes. The aim of our work is to achieve reasonably accurate sketching and seamless switch between 2D sketching and 3D sketching at a low cost using consumer-grade VR trackers and without requiring additional hand-held devices.

3 FORMATIVE STUDY

Several studies already explored the VR sketching task with tablets or hand-held controllers [3, 13, 47]. However, the possibility of using the hands as a physical support for sketching in VR is still largely unexplored. In this section, we present a formative study that we conducted to gain insights into how people envision exploiting free hands or gestures in a VR sketching system. We then leverage the results of this study to build a design space for the tackled problem [13, 22].

3.1 Subjects and Procedure

Fourteen people (6 males, 8 females, aged 22-30), later referred to as P1–P14, were recruited to join the formative study. All of them were new to VR sketching. Some of them (P1–P5) had some experience in 3D modeling software like 3DS Max and Maya.

We split the procedure of the study into two parts. Like in the work by Arora et al. [3], in the first part (organized as a 60-minute session) the subjects were told to sketch freely in the following three modes: on a tablet with a pen, in mid-air with a pen, and on the non-dominant hand with a finger of the dominant hand. Each subject created 184 strokes, on average. While drawing, the subjects were requested to stand. In VR sketching, 2D sketches tend to be integrated into 3D sketches (e.g., in [13]), and 3D sketching requires frequent checking of results from different viewpoints. These operations are expected to be more convenient to perform from a standing posture.

In the second part, the subjects were invited to fill in the questionnaire reported in Figure 3 (right), using a 5-point Likert scale (with 1 representing "strongly disagree" and 5 representing "strongly agree"). To analyze the assigned scores, we ran a Wilcoxon signed-rank test. A semi-structured interview (μ=34.9mins, σ= 8.8mins) was also conducted to investigate the subjects' a) experience with sketching software and VR applications; b) feelings about sketching on a tablet, a hand, and in mid-air; c) expectations about sketching in a VR environment; d) ideas about how to take advantage of the hardware used and the different interaction modalities experienced in the first part to realize the expectations of VR sketching. During the interview, the subjects were allowed to freely use the provided hardware to illustrate and validate their ideas. Interviews with all the subjects produced lots of repeated answers, showing some form of saturation. We obtained a general understanding of their feelings and expectations about VR sketching through a thematic analysis of taken notes and recordings.

(a) (b)

Figure 2: Hardware used in the formative study. (a) Manus VR gloves and HTC Vive trackers, (b) Wacom Intuos Draw digital tablet and pen, with buttons extracted from a slide presenter pen and OptiTrack M4 markers attached to a custom 3D printed case.

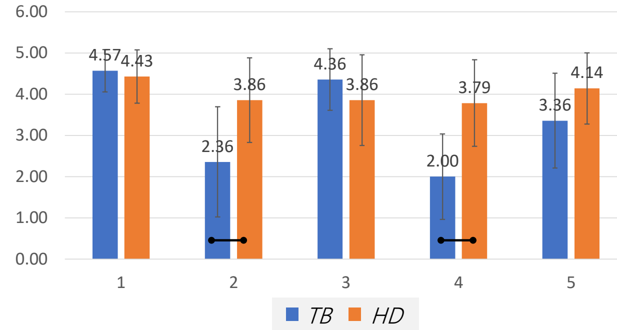
3.2 Apparatus

The hardware used for sketching on a hand (shortly, *HD*), shown in Figure 2(a), encompassed an HTC Vive Head-Mounted Display (HMD), a pair of Manus VR's 'Prime Haptic' gloves (64g each, 2,990 EUR in total), two HTC Vive trackers (89g each, 238 EUR in total) and their associated Lighthouse tracking system. Sketching on a tablet (shortly, *TB*) was implemented using a marker-equipped Wacom Intuos Draw tablet and its pen, shown in Figure 2(b) (290g and 9g, respectively, or 318g and 41g, including an OptiTrack M4 set of markers, 99.99 USD for the tablet and pen set, 14,706 EUR for an OptiTrack tracking system with 12 Flex 13 cameras). The local coordinates of the pen are gathered by the electromagnetic resonance (EMR) sensors of the tablet. The software for sketching both on HD and TB was developed in C# using the Unity game engine, SteamVR SDK, and ManusVR SDK.

3.3 Observations

By observing the subjects' behaviors in the drawing session and collecting their opinions through the questionnaires and the interviews, we drafted the following observations (O1–O4 about sketching on a tablet vs. sketching on a hand, and O5–O8 regarding general aspects of the sketching experience), which guided the definition of our design space.

O1 Physical support. It is challenging to create precise strokes by sketching in mid-air due to the lack of physical support [3]. After sketching with the three modes, the subjects felt that, for generating 2D constrained strokes, both sketching on a tablet and sketching on a hand performed better than sketching in mid-air, as reflected in the results for Q1 in Figure 3. All the subjects agreed



Questionnaire

Q1. Compared to mid-air sketching, sketching on TB/HD is better for generating 2D constrained strokes.

Q2. I think that I would like to sketch on TB/HD if the working time is over 30 minutes.

Q3. I think that using TB/HD for drawing is natural.

Q4. I felt fatigues less easily when sketching on TB/HD.

Q5. I would imagine that most people would learn to sketch on TB/HD easily.

Figure 3: Average scores obtained for each statement in the questionnaire during the formative study. Mean values and standard deviations are expressed via bar heights and error bars, respectively. A line connecting two bars indicates a statistically significant difference between the two scores ($p < .05$).

that sketching in mid-air offered more freedom; however, excessive degrees of freedom also brought the challenge of precise sketching, and sketching on a tablet or a hand provide physical support to alleviate accuracy problems. The score of sketching on a tablet is slightly higher than sketching on a hand, though the Wilcoxon test did not find the difference statistically significant (TB: 4.57 vs HD: 4.43, $p = .5877$). This result was mainly due to the fact that a few subjects thought the hand was not as flat as the tablet, and the drawing region on the hand was also smaller.

O2 Input modes. Pen and touch are both popular interaction methods, which are often adopted by systems with direct inputs [1, 2, 23]. The subjects felt that using either a pen or an index finger is appropriate for drawing (Q3 in Figure 3). Since almost all the subjects use a pen to draw on paper in their daily life, they considered the pen as a natural way for sketching in VR. As for the finger-based interaction, all the subjects had previous experience in sketching by with an index finger on devices with multi-touch screens, like the Apple iPhone, iPad and Microsoft Surface. Initially, a few subjects showed concerns on fat-finger issues [51]. Nevertheless, after trying hand-based VR sketching, they found that such issues were not serious, since the drawing was not occluded by their fingers.

O3 Fatigue. As reflected by the results for Q4 in Figure 3, sketching on a tablet was more fatiguing than sketching on a hand (TB: 2.00 vs HD: 3.79, $p < .05$). P7 commented: "When the time of sketching increased, it was more and more difficult to hold the tablet stably and sketch on it". As anticipated, a quantitative study conducted in [39] suggests that the comfortable holding time of a middle-size tablet (446g) and a small-sized tablet (241g) without performing any other operation is less than 27 minutes and 36 minutes, respectively [39]. After sketching on the selected tablet for 20 minutes, more than half of the subjects reported to feel tired in our study. Should the working time be over half an hour (Q2 in Figure 3), the subjects would significantly prefer sketching on a hand than on a tablet (HD: 3.86 vs TB: 2.36, $p < .05$). P11 commented: "It was challenging to hold a tablet stably when drawing. Each time the pen touched the tablet, the tablet shook a little. Another point was, as the time went by, the accuracy of sketching decreased because of fatigue".

O4 User preference. The subjects believed that most people would learn to sketch either on a hand or a tablet easily (Q5 in Figure 3). No significant difference was found between the considered alternatives. When drawing multiple strokes, we noticed that some users preferred to sketch a new stroke connecting it to existing strokes, like when drawing a petal from a stamen. Some subjects initially thought that sketching on a tablet could be like a natural extension of drawing on a physical canvas. However, after trying this mode in the VR environment, they realized that it might take some time to adapt to drawing on a tablet held by hands.

O5 Sketching expectations. Almost all the subjects agreed that both stroke lines and B-rep 3D surfaces should be available in a VR sketching system. P2 commented: "I hoped to create real 3D objects instead of just strokes".

O6 Sketching operations. It is not easy to create desired strokes in one pass. Thus, editing functionalities are essential for a VR sketching system. Auxiliary functionalities (e.g., color and brush size setting, rescale, creation, de-/selection, transform, copy, delete, and undo operations) are required too.

O7 GUI and gestures. The subjects suggested using gestures to fulfill highly-repeated operations, such as edit and transform, and exploiting a GUI with larger graphics icons to manage the remaining functionalities, since accurate selection in VR is not easy.

O8 Hand-held and wearable devices. All the subjects preferred not to hold or wear any device when sketching or modeling in a VR environment. However, the current optical technologies could not perform robust tracking of two bare hands and accurate recognition of complex interactions [35]. Anyway, the subjects felt that wearable devices were more comfortable than hand-held devices.

3.4 Challenges

From a systematic analysis of the subjects' behaviors and opinions, we identified several challenges for achieving hand-based 3D sketching in VR.

Limited region. Using a hand as a canvas brings the problem of a limited workspace for drawing. As a result, it is challenging for users to create large objects with hand-based sketching.

Fingertip. The degree of sensitivity of drawing individual strokes depends on the real-time position of the drawing hands' index fingertip in the canvas hand's drawing region recorded by the hardware. Most of the available digital gloves relying on flex sensors can only measure the tensions on the bending joints and convert them to spatial coordinates. For example, the Manus VR gloves exploited in the study provide real-time coordinates of 16 points (Figure 6(a)) on the hand (the first, second, and third joint points for each finger and the bottom point of the palm) in its local frame. However, the fingertips are not tracked by the gloves.

Uneven canvas. A hand is not as flat as a tablet because of the existence of the spaces between fingers. While sketching on a hand, it may be difficult for users to keep it as a perfect plane (or some specific parametric surface). On the one side, we intend to sketch planar curves, which require an ideal plane proxy. On the other side, we need a trigger to (de)activate the drawing process automatically by detecting whether the fingertip touches the canvas hand or not. The trigger should match the current hand shape. Therefore, how to define an appropriate trigger volume is a challenging issue.

Jittering. When drawing on a hand, it is not easy for users to make the hand-based canvas stable, since hands are neither fixed nor rigid. As a result, the tracked global coordinates of the non-dominant hand returned by the hardware will be jittering, making sketches based on the raw data suffer from visible artifacts. For example, we frequently observed a gap between two strokes that should have been connected, or strokes that include zigzags beyond users' intentions.

3.5 Design Space Dimensions

For each candidate dimension of the design space, potential questions and intriguing considerations are put forward to envision the role of hands in VR sketching tasks and corresponding interactions. It is worth recalling that we focus on establishing a design space and designing interactions for using bare hands as input in VR sketching tasks.

D1 Physical proxy vs. Visual guidance. As said, the accuracy of VR sketching in mid-air is much lower than that of traditional 2D sketching. The use of a physical proxy and visual guidance could compensate for this loss [3]. Which one of these approaches should be adopted in a VR sketching system (or are both of them supposed to be integrated)?

D2 Physical surface types. Both a tablet [2, 13] and a smartphone [32] have already been adopted to provide planar physical surfaces. Apart from planar surfaces, could a hand provide other types of physical surfaces in VR sketching tasks?

D3 Object types. 3D shapes include billboards, B-rep objects, etc. Concept design and brainstorming tasks tend to adopt billboards to express ideas, whereas B-rep objects are more commonly used for product and industrial design. Should a VR sketching system offer either one of them or both?

D4 Uni-modal vs. Multi-modal interaction. Existing interaction methods such as GUI [13], gestures [47], gaze [40], etc., have been adopted into VR environments for enabling interactions. How could these methods be applied to reduce sketching inaccuracy and provide, at the same time, intuitive control over the different functionalities of a VR sketching system?

D5 Unimanual vs. Bimanual interaction. Although bimanual interaction has lots of benefits [19], it might not suit every task [47]. When mapping operations to gestures, should a gesture be executed by one hand or both hands?

D6 Hand assignment. How to distribute different operations, such as sketching, transform, edit, etc., to the non-dominant and dominant hands to offer natural interaction? For instance, should a user exploit the index finger of the dominant hand to sketch on the non-dominant hand or vice versa?

D7 Direct vs. Indirect interaction. An operation could be fulfilled with either direct inputs or indirect inputs. For instance, a transform operation could be executed either by directly dragging a selected object with a cursor or moving it remotely with a tablet by using the touch capabilities of the device. How to distribute the use of direct and indirect interactions onto operations?

D8 Discrete vs. Continuous input. A task could be implemented through combinations of discrete and continuous inputs. For instance, for the copy operation, discrete input would be more appropriate. How to assign discrete or continuous inputs to various operations?

D9 Interleaved vs. Simultaneous input. Does an operation require simultaneous multi-modal inputs, such as performing a gesture and touching a GUI icon at the same time, or could it be implemented by interleaving a series of uni-modal inputs? Which alternative is more effective in reducing the time for completing the operation and is more natural to accomplish?

D10 Sitting vs. Standing. Body postures could affect users' interaction experience, especially in a VR environment [21]. Does any functionality of a VR sketching systems require users to move in the VR environment? Should a VR sketching system offer alternative operating conditions (e.g., sitting or standing)? Could system functionalities introduce constraints regarding the operating conditions?

4 SYSTEM COMPONENTS

In this work, we present a novel 3D sketching system that uses the non-dominant hand as a canvas, and an index finger from the dominant hand as a pen. Our system facilitates the task of creating planar/surface constrained strokes, 3D mid-air free strokes, and 3D B-rep surfaces, which can be edited and fused together to compose various artworks or geometric designs in a VR environment.

The hardware of our system is composed of a VR HMD, two VR gloves that can provide precise finger tracking of each hand, and a motion tracking accessory mounted on each glove to track its global pose. In our experimental prototype, we adopted the same devices used in the formative study. By wearing the gloves, the user can draw on the non-dominant hand by using the tip of the index finger of the dominant hand as the input. The 3D positions and orientations of the palm canvas and the drawing finger are tracked in real-time. The system creates and displays both strokes and B-rep surfaces defined by the user's input. Finally, the user can exploit hand-based interactions to further manipulate selected strokes, e.g., translating, rotating, scaling, editing, or copying them.

We achieve the above-mentioned functions by different gestures and GUI (see Figure 4 (Bottom)). Gesture recognition is performed based on sensors (proximal flex sensor data, and medial sensor data from each finger) integrated in the gloves. Our prototype was implemented and successively tested by using an Alienware desktop PC equipped with an Intel i7-8700K and an NVIDIA GeForce GTX 1070. The software of our system was developed in C# using the Unity game engine, SteamVR SDK, and ManusVR SDK. Please refer to the accompanying video for 3D sketching sessions with our system.

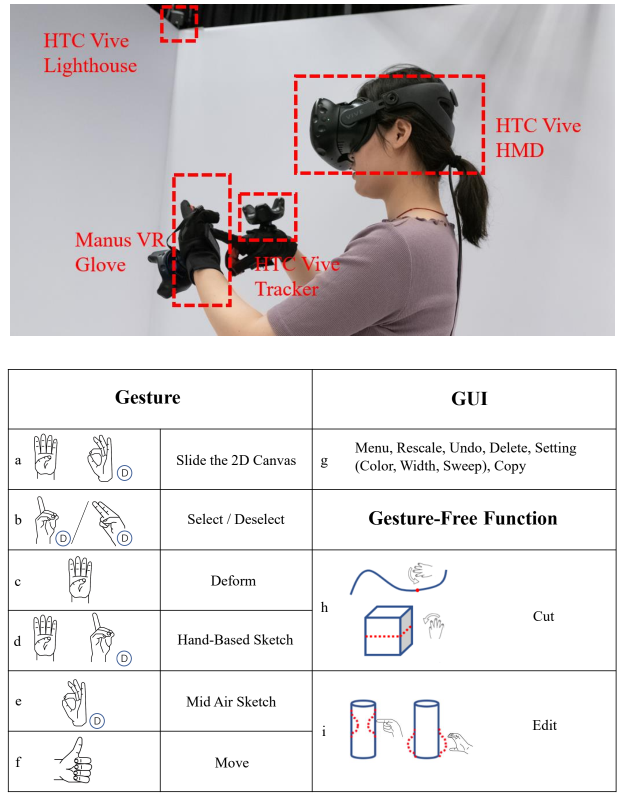


Figure 4: (Top) The hardware of our system, including an HTC Vive HMD, two Manus VR's 'Prime Haptic' Gloves, two HTC Vive trackers and their associated Lighthouse tracking system. (Bottom) Functions implemented in our system. This illustration assumes the use of the left hand as the canvas hand and the right hand as the drawing hand. (a)-(f) functions triggered by gestures; (g) functions triggered by GUI; (h) cut operation; (i) edit operation. Both (h) and (i) are gesture-free operations. Red lines in (h) show a cutting point or a cutting intersection line, whereas in (i) show the target shape of a B-rep surfaces after editing.

4.1 System Design

Based on the findings of the formative study, our system (Figure 4 (Bottom)) offers two operation modes, namely the creation mode and the editing mode. According to D3, a VR sketching system should support generating B-rep objects to improve the modeling capabilities of systems that use billboard objects only. Thus, the creation mode includes three sub-modes: 2D constrained stroke creation by sketching on a hand, 3D mid-air free stroke creation by sketching in mid-air, and 3D B-rep surface creation by a sweep operation. All the strokes are rendered as tubes (generalized cylinders) overlaid on top of the user' fingers to address occlusion issues caused by the fat-finger problem. In the editing mode, the user can manipulate both selected strokes and B-rep surfaces using various operations, including scaling, deformation, cutting, positioning, etc. Taking D4 into consideration, our system adopts multi-modal interactions, including gestures and GUI for frequent operations (e.g., hand-based sketching) and fine adjustments (e.g., rescaling) respectively. We designed gestures to use in our system based on D5–D9. Based on D5, bimanual interaction was used for operations requesting two hands like, e.g., hand-based sketching and 2D canvas sliding. According to D6, the dominant hand was assigned to common operations, such as menu opening and selection. Concerning D7, besides selection and sketching, all the other basic operations were realized by indirect interaction. As for D8, discrete operations, such as undoing and deleting, were fulfilled by discrete triggers, whereas continuous functions, such as rescaling and editing, were realized by continuous adjustments. With respect to D9, operations with simultaneous multi-modal input, such as rescaling and duplicating, could be better fulfilled by GUI. Below we give more details of the key features of our system.

4.1.1 Creation Mode.

This mode supports interactive creation of 2D constrained strokes, 3D free strokes, and B-rep surfaces (according to D3).

Sketching on a hand. Our system allows users to use an index finger from the dominant hand (called the drawing hand) to draw on the non-dominant hand (called the canvas hand), regarded as a physical proxy [3], to create 2D constrained strokes. A simple but easy-to-keep gesture, i.e., bending the thumb of the canvas hand (Figure 4 (Bottom), gesture (d)), is used to activate the canvas hand. All the newly generated strokes will be attached to the activated canvas hand. When the canvas hand is deactivated, the strokes will become detached and will stop moving in the virtual space. When the canvas is being activated, the system detects whether the fingertip of an index finger from the drawing hand touches the trigger volume (Figure 6(d)) or not, in order to start or end an individual sketch line accordingly.

Sketching in mid-air. Our system also supports mid-air drawing. When holding a pinch gesture (Figure 4 (Bottom), gesture (e)), the hand motion trajectory in mid-air is used to generate a 3D stroke.

Creating B-rep surfaces. We sweep a B-rep object by using a surface-constrained stroke (Figure 5(a), created with the hand-based sketching modality) as the section and a 3D stroke (Figure 5(b), generated with the mid-air sketching modality) as the guideline. To improve the quality of created B-rep surfaces, we remove unintended self-intersections of a closed section line by computing its concave hull.

4.1.2 Editing Mode.

In this mode, our system offers several basic functions: positioning, copy, deforming, cutting, editing, rescaling, deleting, undo, and color and width setting. Each function is activated by a corresponding hand gesture or a GUI button, as summarized in Figure 4 (Bottom).

Positioning. This operation is used to translate or rotate selected objects to an appropriate location. It can be triggered by a thumb-up gesture of the canvas hand (Figure 4 (Bottom), gesture (f)). It transforms the selected objects according to the movement and orientation of the canvas hand.

Deforming. This operation lets users deform selected 2D constrained strokes in order to generate 3D spatial curves. An straightforward approach for creating spatial curves is to sketch on a bent

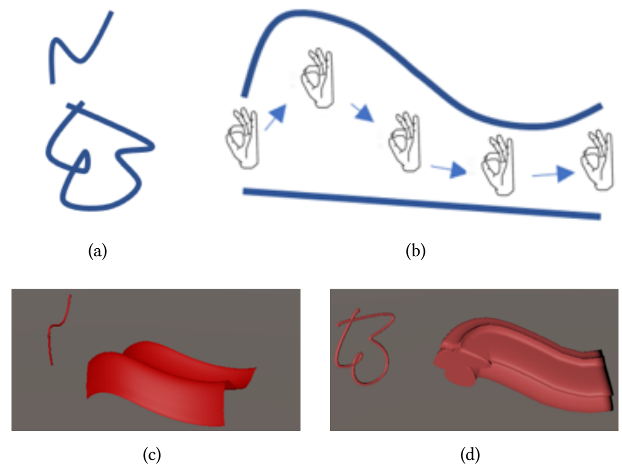


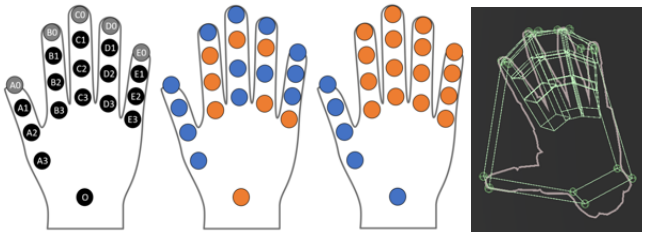
Figure 5: The process of generating B-rep objects. (a) and (b) show the generation process: (a) two examples of section lines; (b) sweep operation to generate sweep lines. (c) and (d) show two examples of generated B-rep objects and the corresponding section lines.

hand directly. However, it is hard to stably track the joints of a bent hand while sketching on it. From the formative study, we also observed that the users preferred to edit 2D strokes after having sketched them, instead of creating spatial curves in one pass (O6). We thus adopted a sketch-and-deform approach. To deform selected 2D strokes, the user bends the thumb of the drawing hand (Figure 4 (Bottom), gesture (c)). The 2D constrained strokes are projected onto a NURBS proxy, which is dynamically driven by 16 control points (Figure 6(c)) pre-defined on the non-dominant hand.

Editing. The editing operation allows users to edit the shape of B-rep surfaces. Inspired by a Leap Motion-based interaction method [43], we exploited a gesture-free geometric approach to edit the shape of objects, as illustrated in Figure 4 (Bottom), gesture (i). When only one finger pushes a swept surface, its section radius is decreased accordingly. When two fingers push the surface, the section radius of the part between two touch points is increased.

Cutting. Either a tablet or a hand, regarded as a physical "knife", has been leveraged to "cut" objects [16, 47]. We adopted a similar approach and used the palm of either hand as a physical knife to cut strokes or B-rep surfaces. This gesture-free operation cuts off the smaller part of an object and keeps the larger part, as illustrated in Figure 4 (Bottom), gesture (h).

Auxiliary operations. Our system also supports additional operations including copying, rescaling, deleting, undo, and color/width settings with the assistance of a GUI menu [8]. A button shown in the virtual environment can be used to open the GUI menu during the whole sketching process. Our system also offers an auto-snapping function, since it is not easy for users to accurately perceive depths and poses of objects in the 3D world [21].



(a) (b) (c) (d)

Figure 6: Hand point positions used in our system. (a) Hand point positions captured by the Manus VR gloves (in black) and computed by our system (in grey); (b) Points selected for plane fitting (in orange); (c) Points selected for NURBS fitting (in orange). (d) The trigger bounding polyhedrons used to detect whether the user is drawing or not.

4.2 Solutions to Identified Challenges

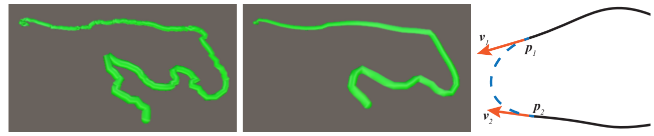
To solve the challenges identified from the formative study, we put forward the corresponding solutions as follows.

Limited region solution. Due to the limited size of the hand-based canvas, the sketching input region is constrained. To expand this region, we allow users to pan the canvas by using a pinch gesture of the drawing hand (see Figure 4 (Bottom), gesture (a)).

Fingertip solution. By relying on the data about the global frame provided by the HTC Vive trackers, our system easily converts all the local positions on hands to the world coordinates. To obtain the position and orientation of each fingertip (the grey points in Figure 6(a)), a calibration step is needed requesting a user to stretch out all the fingers of one hand one by one to make them touch the other hand perpendicularly several times, so that the system can calculate the distance of each fingertip to the corresponding first joint point.

Uneven canvas solution. To locate the drawing region, we consider the canvas hand as planar proxy, fitted in a least-squares manner by selecting a stable subset of points from the canvas hand. In this case, calibration is achieved by posing the hand to make it represent an ideal plane. Then we select 8 points with the minimal distances to the fitted plane as shown in orange in Figure 6(b). We noticed that the current non-dominant hand deviates slightly from the fitted plane. To deal with such deviations, we adopted a polyhedron proxy to approximate the hand shape. Specifically, we separate the hand into several sub-regions, including the palm plane and finger parts. Then, we generate a bounding polyhedron centered on the palm proxy with a thickness of 3cm. We build similar bounding polyhedrons with the same thickness to describe the finger parts. As illustrated in Figure 6(d), these elements define a stable trigger volume for drawing.

Jittering solution. To alleviate jittering artifacts like those exemplified in Figure 7(a), we adopt Kalman filtering and dynamic uneven resample algorithms [12, 41] for smoothing individual strokes. See Figure 7(b) for the processed strokes. To generate better connected strokes, our system supports the automatic connection of two strokes by generating cubic Bézier curves if the Euclidean distance between two endpoints, one from the current stroke and the



(a) (b) (c)

Figure 7: Strokes beautification: (a) and (b) are the strokes before and after the application of dynamic uneven resampling and Kalman filtering. (c) Bézier curve (blue dashed line) generated to smoothly connect two close strokes (black solid lines).

other from an existing stroke, is below a certain threshold (2cm in our system). To decide the control polygon of a cubic Bézier curve, as shown in Figure 7(c), we calculate the tangent vectors $\mathbf{v}_1$ and $\mathbf{v}_2$ for the two endpoints to be connected $\mathbf{p}_1$ and $\mathbf{p}_2$, and approximate the curvature c_1 and c_2, respectively, by a segment-length weighted Laplacian operator [37]. Then, we set the length of $\mathbf{v}_1$ and $\mathbf{v}_2$ as $\ell_i = \frac{1}{c_i+\epsilon}$ (with trimming to avoid intersection), where ϵ is a tiny constant value. As a result, the two strokes are connected smoothly. Compared with an alternative approach of moving or re-shaping an entire stroke to connect it with other strokes, our method preserves user-created strokes better.

5 EVALUATION AND RESULTS

5.1 Evaluation

To investigate effectiveness and the usability of the proposed system, we conducted a user study, in which we compared the performance in terms of drawing time, overall accuracy, and robustness of our system for sketching in a VR environment with an HMD on a hand (HD for short) with that of sketching on a tablet (TB for short) and of Google Tilt Brush[1] (GTB for short). GTB was selected to represent state-of-the-art VR-based 3D sketching/painting software.

Figure 10 illustrates the three systems in use. The hardware and software configuration used for the HD and TB was the same adopted in the formative study. The GTB setup leveraged an HTC Vive controller tracked with the Lighthouse tracking system.

The stroke cross-section geometry, width, and color, as well as the feedback and the environment map of HD and TB were set up to be as similar as possible to GTB for fair comparisons. Given the fact that the mechanism of creating freeform strokes in a 3D space with the three systems is similar, our user study focused on evaluating their performance on drawing 2D sketches in VR, which requires quite different interactions.

Participants. 12 volunteers (5 males and 7 females) later referred to as P1–P12, aged between 23 and 29 (μ=25.25, σ= 1.60) participated in the study. All of them were right-handed, and had no experience with 3D sketching in VR. We chose to focus on novice users (identified through a set of demographic questions aimed to understand the subjects' expertise with technologies related to the experiments, i.e., VR systems, 3D modeling and sketching tools) in order to assess the intuitiveness of the considered systems without possible biases due to prior knowledge and skills.

[1] https://www.tiltbrush.com/

Procedures. Inspired by [3] and [32], we asked each subject to perform three tasks using HD, TB, and GTB while standing [13]. (1) The "Circle" task encompassed the drawing of a perfect 2D circle. This task was aimed to assess whether the subjects could draw a desired shape well. (2) The "Link" task requested the subjects to draw a 2D straight line connecting two virtual points. This task was repeated multiple times to create lines distributed evenly in the horizontal and vertical directions. (3) The "Ladder" task encompassed the drawing of a 4-step ladder. This task was designed to test the ability of the subjects to create a sketch involving multiple, well-connected strokes. The order of the three systems being compared was counterbalanced by a Latin square design to minimize possible learning effects. Each drawing session with a single system lasted approximately 35 minutes. Before passing from one system to the other, the subjects were asked to take a break for at least 20 minutes. In a preliminary 5-minute tutorial session, the subjects learnt how to draw with each system and were introduced to the tasks. At t = 5mins, 10mins, 15mins, 20mins, 25mins, and 30mins, the participants were required to draw a circle 5 times and link two points 5 times (to compare the drawing error in the three systems), and draw a ladder once (for collecting the drawing time). Every 5 minutes, the subjects were requested to assign a score in a 1-to-9 scale to their degree of fatigue (with 1 meaning "not fatigued at all", 9 meaning "severely fatigued"). In the spare time between two time points, they were requested to draw some simple objects (like a house, a tree, or a flower) chosen by the experiment administrator for keeping the subjects drawing for the whole session. After completing the three sessions, the subjects were requested to fill in a post-test questionnaire aimed to evaluate the performance of the three systems from a subjective perspective.

5.2 Evaluation Criteria

To measure the performance of the three systems, we considered both objective and subjective metrics. As for objective metrics, a drawing error was calculated as the averaged minimal distances between points sampled along the strokes and the target shape proxies. Moreover, the time requested to complete the "Ladder" task was also collected. The subjective evaluation was based on the fatigue levels collected during the experiment and the results of the post-test questionnaire. The questionnaire, which is included in the supplemental material, was made up of two sections. The first section aimed to evaluate the usability of the three systems by using the System Usability Scale (SUS) [7]. In the second section, the subjects were requested to evaluate specific usability factors of the systems, i.e., user input, functionality, flexibility, and overall usability according to the statements proposed in the VRUSE tool [25].

5.3 Results and Feedback

A qualitative overview of the drawing results is provided in Figure 8 and Figure 9. In particular, Figure 8 shows the overlapped sketches created by all the subjects in the "Circle" and "Link" tasks, whereas Figure 9 illustrates representative sketches for the "Ladder" task obtained at different time intervals. In the following, the three systems are compared in quantitative terms using the results obtained by applying the evaluation criteria described above.

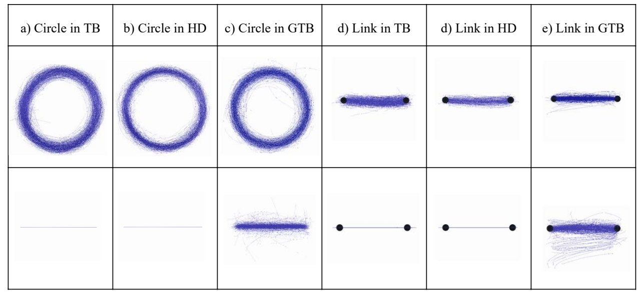


Figure 8: Overlapped results from all the subjects in the "Circle" and "Link" tasks for the three systems.

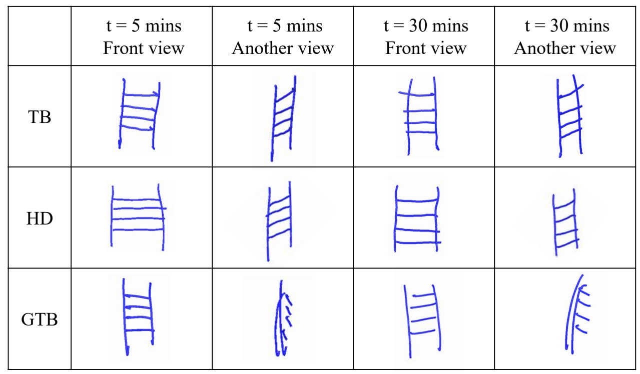


Figure 9: Representative results from different subjects in the "Ladder" task for the three systems.

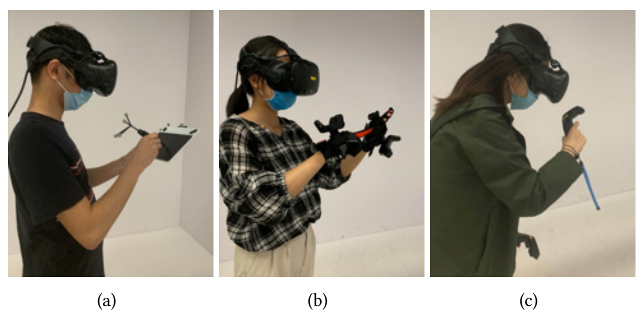
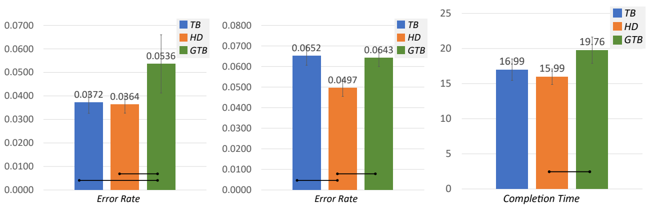


Figure 10: Participants involved in the user study drawing with the a) TB, b) HD, and c) GTB systems.

5.3.1 Objective Results.

On average, each subject created 279 strokes for the "Circle", "Link" and "Ladder" tasks. Figure 11 shows the statistics for objective metrics. Statistical significance of the results was analyzed by a one-way ANOVA test, followed by Paired Student's t-tests with the Bonferroni correction for pairwise comparisons.

Considering statistically significant results and starting from the error rate measured for the "Link" task (shown in Figure 11(a)), it can be observed that with HD and TB the subjects were more accurate than with GTB ($F_{(2,33)}$ = 4.54, GTB: .0536 vs HD: .0364,



(a) Link Task (b) Circle Task (c) Ladder Task

Figure 11: Objective results. a) Error rate for the "Link" task, b) error rate for the "Circle" task, and c) completion time for the "Ladder" task. Mean values and standard deviations are expressed via bar heights and error bars, respectively. A line connecting two bars indicates a statistically significant difference between the corresponding systems ($p < .05$).

$p = .0378$, and TB: .0372, $p = .0487$). This result could be explained by the presence of the physical support that helped the subjects to draw more accurate lines. When dealing with the "Circle" task (Figure 11(b)), HD let the subjects achieve significantly better results compared to both TB and GTB ($F_{(2,33)} = 12.02$, HD: .0497 vs TB: .0652, $p = .0002$, and GTB: .0643, $p = .0003$). This result is also confirmed by the feedback provided by the subjects at the end of the experiment. In fact, P3 commented: "At the beginning, sketching on a tablet helped me to create 2D circles accurately. However, later on, it was more and more difficult for me to hold it stably, especially when the pen was touched". Compared with TB and GTB, all the subjects agreed that HD could be helpful for 2D sketching for a longer time.

For the completion time of the "Ladder" task (depicted in Figure 11(c)), it can be noticed that, with HD, the subjects were significantly faster ($F_{(2,33)} = 4.89$) than with GTB (15.99s vs 19.76s, $p = .0076$). Without physical support, it is hard to accurately connect the steps of a ladder. P7 commented: "When sketching in mid-air with GTB, I had to spend more time checking if two strokes were connected or not. Another point was that after I created a ladder, I found the steps to be not straight, they were more like 3D curved lines". Another comment, by P10, was: "As time went by, the difficulty associated with drawing a ladder with TB increased. After around half of a session, I was fully exhausted and needed to spare no effort to finish the tasks." The difficulties highlighted by the subjects can be spotted also in Figure 9, since at both t = 5mins and t = 30mins the ladders created with TB show strokes with messy start and end points.

5.3.2 *Subjective Results.*

Statistical significance of the subjective results was analyzed by the Friedman's test ($p < .05$), followed by the Wilcoxon Signed-Rank test for pairwise comparisons.

From the results concerning the degree of fatigue (Figure 12), it can be observed ($Q = 16.87$, $p = .0002$) that TB (4.82) was the system that made the subjects perceive higher fatigue for operations that lasted a long time with respect to both HD (2.83, $p = .0022$) and GTB (3.19, $p = .0044$). This result is confirmed by the subjects' comments. For instance, P4 stated that: "Fatigue of GTB and HD was acceptable. But, when drawing with TB, after holding the tablet for 15-20 minutes my hand started to shake a little. This resulted in a bit more effort to finish the tasks".

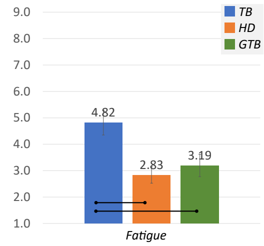


Figure 12: Fatigue perceived during the experiment. Mean values and standard deviations are expressed via bar heights and error bars, respectively. A line connecting two bars indicates a statistically significant difference between the corresponding systems ($p < .05$).

Table 1: Subjective results about the overall usability according to SUS [7].

System	Score	Grade	Adj. Rating
TB	46.67	F	Poor
HD	71.46	C+	Ok
GTB	43.33	F	Poor

Table 1 reports the SUS scores of the compared systems. It can be noticed ($Q = 11.17$, $p = .0037$) that the subjects perceived HD as characterized by higher usability (71.45) than both TB (46.67, $p = .0038$) and GTB (43.33, $p = .0029$). No statistically significant difference was observed between TB and GTB. According to the categorization in [4], HD was rated as grade C+, whereas TB and GTB both obtained an F grade. We speculate that the significant difference found in terms of usability between HD and the other two systems was due to the higher fatigue and the difficulties the subjects had in drawing accurate lines close to the end of the session. Although the subjects attempted to maintain their drawing accuracy high, the results were not as expected because of the tiredness of the hand holding the tablet (with TB) or because of the issues related to mid-air drawing (with GTB). For instance, in the "Link" task, many subjects often found that the connection was almost perfect in one view, but bad in another view. Thus, they might have felt disappointed at the end of the experiment because of the poor results obtained in this session. P2 commented: "GTB defeated me. No matter how much effort I put on the sketching, the strokes were either floating or not connected. I was disappointed with the drawing results obtained using that system".

Regarding the second section of the questionnaire, from the overall scores reported in Figure 13, it can be observed that, in general, HD outperformed the other two systems with respect to all the usability factors considered. More specifically, statistically significant differences were found between HD and the other two systems for what it concerns functionality ($Q = 8.97$, $p = .0112$; HD: 4.08 vs TB: 2.92, $p = .0357$, and GTB: 2.92, $p = .0125$), flexibility ($Q = 12.64$, $p = .0018$; HD: 4.08 vs TB: 2.67, $p = .0077$, and GTB: 3.25, $p = .0277$), and system usability ($Q = 9.65$, $p = .0080$; HD:

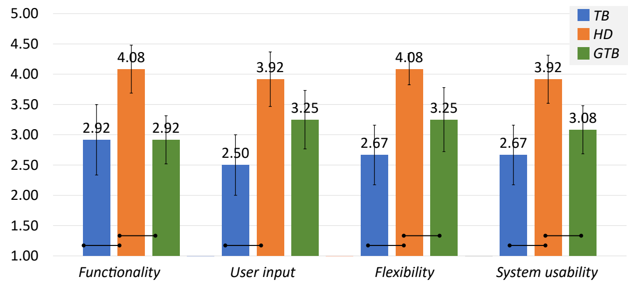


Figure 13: Overall scores for the usability factors according to VRUSE [25]. Mean values and standard deviations are expressed via bar heights and error bars, respectively. A line connecting two bars indicates a statistically significant difference between the corresponding systems ($p < .05$).

3.92 vs TB: 2.67, p = .0180, and GTB: 3.08, p = .0218). For the user input, a significant difference (Q = 9.69, p = .0078) was found only between TB and HD (2.50 vs 3.96, p = .0088).

It is possible to further analyze, with the same strategy adopted above, the scores assigned to questions characterizing each usability factor. In particular, the subjects found that HD was characterized by a degree of functionality higher than TB and GTB (Q = 11.53, p = .0031; HD: 4.25 vs TB: 3.00, p = .0117, and GTB: 2.50, p = .0180). The functioning of HD was also judged (Q = 8.42, p = .0148) as less ambiguous (2.00) than GTB (3.00, p = .0117).

With respect to the user input factor, HD was rated as easier to use than TB (Q = 8.72, p = .0127; HD: 3.75 vs TB: 2.42, p = .0108). The subjects found HD and GTB more adequate for the tasks, thus perceiving less need for an alternative interface less than with TB (Q = 6.37, p = .0414; TB: 3.92 vs HD: 2.50, p = .0151, and GTB: 3.00, p = .0381). Moreover, HD was perceived as more ideal for interacting with the virtual environment than both TB and GTB (Q = 13.85, p = .0009; HD: 4.00 vs TB: 2.42, p = .0050, and GTB: 2.42, p = .0117), as well as the system that made the subjects feel to have more control over the operations to be performed (Q = 11.70, p = .0029; HD: 4.08 vs TB: 2.58, p = .0117, and GTB: 2.50, p = .0180). Finally, with HD, the subjects had the perception of making a lower number of errors than with GTB (Q = 7.31, p = .0258; HD: 2.58 vs GTB: 3.58, p = .0280).

Concerning the flexibility factor, the subjects judged HD as the system that better succeeded in letting them perform what they actually wanted to do (Q = 13.61, p = .0011; HD: 1.92 vs TB: 2.75, p = .0277, and GTB: 3.75, p = .0117); it was also found to give them the possibility to take more shortcuts than GTB (Q = 8.00, p = .0183; HD: 3.50 vs GTB: 2.42, p = .0277).

Lastly, as for the system usability factor, with TB the subjects had the feeling that the system worked against them more than with HD (Q = 7.05, p = .0293; TB: 3.17 vs HD: 2.17, p = .0243). HD was considered more comfortable to use for long periods than both TB and GTB (Q = 7.59, p = .0224; HD: 3.50 vs TB: 2.17, p = .0209, and GTB: 2.00, p = .0152). Moreover, with respect to GTB, HD was judged as less difficult to learn (Q = 7.65, p = .0217; GTB: 2.67 vs HD: 1.75, p = .0357) and capable to make the user feel more in control of the operations (Q = 8.65, p = .0132; GTB: 2.92 vs HD: 4.17, p = .0180). Furthermore, with HD, the subjects had the feeling that the system worked more as expected than with TB and GTB (Q = 8.76, p = .0124; HD: 2.00 vs TB: 3.42, p = .0125, and GTB: 3.58, p = .0300). The subjects found more benefits in the man-machine interaction style offered by HD and GTB than by TB (Q = 8.85, p = .0119; TB: 2.83 vs HD: 3.92, p = .0180, and GTB: 3.50, p = .0277). Lastly, with HD, the subjects found less difficult to work in 3D than with TB and GTB (Q = 10.85, p = .0044; HD: 2.00 vs TB: 3.33 p = .0180, and GTB: 4.00, p = .0108), and HD was considered as more enjoyable than TB (Q = 10.55, p = .0051; HD: 3.58 vs TB: 2.08, p = .0077).

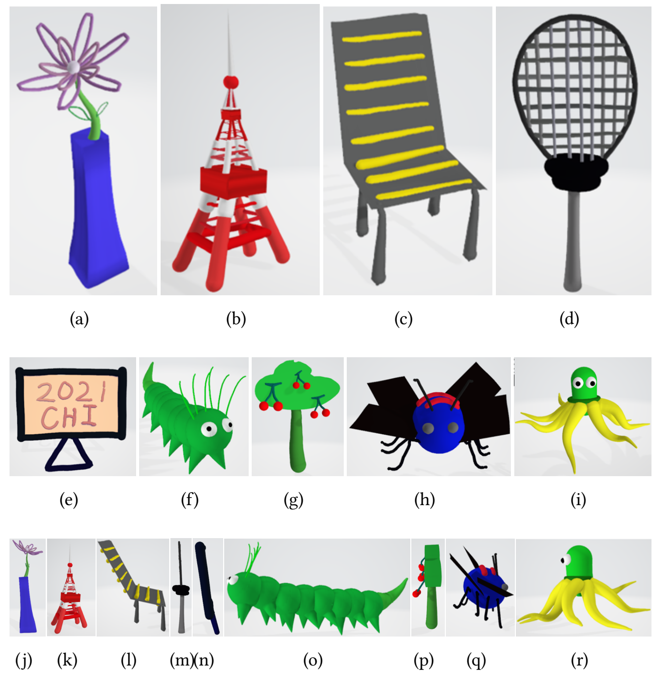


Figure 14: A gallery of drawings made while creating showcases for the proposed system. The drawing time and the number of strokes of each showcase are as follows: a). 4.7 mins, 18 strokes; b). 22.6 mins, 63 strokes; c). 2.9 mins, 15 strokes; (d). 17.4mins, 22 strokes; e). 3.4 mins, 19 strokes; f). 6.2 mins, 35 strokes; g). 2.6 mins, 16 strokes; h). 5.5 mins, 27 strokes; i). 3.9 mins, 14 strokes.

5.4 Expressiveness

Three university students (S1–S3) with good drawing skills were recruited to create showcases for our system and participated in a semi-structured interview (μ=22mins, σ= 3mins) in which they provided their feedback on the experience.

The average training time for each subject was less than 1 hour (μ=41.33mins, σ= 6.11mins). Figure 14 shows the resulting drawings. All the subjects spoke highly of the integration of dexterous hand interaction with 3D VR sketching. S1 commented: "Exploiting fingers and hands to edit the shape of both strokes and 3D shape (meaning the B-rep objects) is intuitive, just like creating pottery or clay". For instance, the flower in a vase in Figure 14(a) was created by editing a B-rep surface and some strokes with fingers. Seamless integration of 2D and 3D sketching enabled the subjects to draw with more freedom than in a single sketching mode. The body of the caterpillar in Figure 14(f) was generated with hand-based sketching, whereas its hair was created with free 3D sketching. All the subjects agreed that physical proxy offered by hands was good for creating hand-based constrained strokes (Figure 14(d)), and our system enabled them to create 3D shapes quickly and intuitively. Finally, several participants agreed with the fact that most of the drawings created with our system would be difficult and time-consuming to create with existing VR sketching software or applications.

6 CONCLUSION AND FUTURE WORK

In this paper, we presented the first VR sketching system with hand-based physical proxy, letting users draw on their hands for both 2D and 3D sketching. Our evaluation suggested that the proposed system allows the study participants to be more accurate with respect to the state-of-the-art solutions, by solving the challenges related to the lack of precise control. Moreover, the subjects involved in the study found the proposed system had highly usability than Google Tilt Brush, and they expressed positive comments about the ease of use and enjoyment. The devised interaction workflow and gestures have the potential to be extended to cope with more general interaction tasks in VR environments with digital gloves.

Although all the subjects appreciated the usability of our system, two of them (P1, P12) stated that drawing in VR without any devices might still be the best solution. Thus, we will explore the possibility of drawing on hands without gloves. Moreover, it was confirmed that the drawing accuracy of VR sketching largely depends on the accuracy of the adopted tracking system. However, existing tracking systems (e.g., OptiTrack or Lighthouse), are sometimes unstable. Although we have applied stroke beautification algorithms to smooth out tracking noise, sometimes beautified results could be inconsistent with the users' intentions in case of heavy jittering effects. The problem of how to beautify strokes without reducing aesthetic quality and affecting users' intentions remains to be addressed. Last but not least, it would be valuable evaluating every function of a VR sketching system separately, in order to explore how they affect the final results.

7 ACKNOWLEDGMENTS

We thank the anonymous reviews for their valuable comments and Mr. Yupu Lu for recording the demonstration demo and the study subjects for their inputs. Especially, we thank Dr. Huaishu Peng for his constructive suggestions on revising this work. This work was partially supported by the General Research Fund project number 17203620 under the Research Grant Council, HKSAR Government and the VR@POLITO initiative of Politecnico di Torino. Hongbo Fu was also supported by an unrestricted gift from Adobe, City University of Hong Kong (No. 7005590, 7005176), and the Centre for Applied Computing and Interactive Media (ACIM) of School of Creative Media, CityU.

REFERENCES

[1] Anand Agarawala and Ravin Balakrishnan. 2006. Keepin'it real: pushing the desktop metaphor with physics, piles and the pen. In *Proceedings of the SIGCHI conference on Human Factors in computing systems*. ACM, Montréal Québec, Canada, 1283–1292.

[2] Rahul Arora, Rubaiat Habib Kazi, Tovi Grossman, George Fitzmaurice, and Karan Singh. 2018. Symbiosissketch: Combining 2d & 3d sketching for designing detailed 3d objects in situ. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. ACM, Montreal, 185.

[3] Rahul Arora, Rubaiat Habib Kazi, Fraser Anderson, Tovi Grossman, Karan Singh, and George W Fitzmaurice. 2017. Experimental Evaluation of Sketching on Surfaces in VR.. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, Vol. 17. CHI, Denver Colorado, USA, 5643–5654.

[4] Aaron Bangor, Philip Kortum, and James Miller. 2009. Determining what individual SUS scores mean: Adding an adjective rating scale. *Journal of Usability Studies* 4 (2009), 114–123.

[5] Mayra Donaji Barrera Machuca, Wolfgang Stuerzlinger, and Paul Asente. 2019. The Effect of Spatial Ability on Immersive 3D Drawing. In *Proceedings of the 2019 on Creativity and Cognition*. ACM, San Diego CA USA, 173–186.

[6] Wendy Bolier. 2017. *Drawing in a virtual 3D space-Introducing VR drawing in elementary school art education*. Master's thesis. Utrecht University.

[7] John Brooke et al. 1996. SUS-A quick and dirty usability scale. *Usability evaluation in industry* 189, 194 (1996), 4–7.

[8] Jeff Butterworth, Andrew Davidson, Stephen Hench, and Marc T Olano. 1992. 3DM: A three dimensional modeler using a head-mounted display. In *Proceedings of the 1992 symposium on Interactive 3D graphics*. ACM, Cambridge Massachusetts, 135–138.

[9] Bruno R De Araùjo, Géry Casiez, and Joaquim A Jorge. 2012. Mockup builder: direct 3D modeling on and above the surface in a continuous interaction space. In *Proceedings of Graphics Interface 2012*. ACM, Toronto Ontario, Canada, 173–180.

[10] Michael F Deering. 1995. HoloSketch: a virtual reality sketching/animation tool. *ACM Transactions on Computer-Human Interaction (TOCHI)* 2, 3 (1995), 220–238.

[11] Ellen Yi-Luen Do. 2001. VR Sketchpad: Create Instant 3D Worlds by Sketching on a Transparent Window. In *Computer Aided Architectural Design Futures 2001*, Bauke de Vries, Jos van Leeuwen, and Henri Achten (Eds.). Springer Netherlands, Dordrecht, 161–172.

[12] David H Douglas and Thomas K Peucker. 1973. Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. *Cartographica: the international journal for geographic information and geovisualization* 10, 2 (1973), 112–122.

[13] Tobias Drey, Jan Gugenheimer, Julian Karlbauer, Maximilian Milo, and Enrico Rukzio. 2020. VRSketchIn: Exploring the Design Space of Pen and Tablet Interaction for 3D Sketching in Virtual Reality. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. ACM, Honolulu, 1–14.

[14] John J Dudley, Hendrik Schuff, and Per Ola Kristensson. 2018. Bare-handed 3D drawing in augmented reality. In *Proceedings of the 2018 Designing Interactive Systems Conference*. ACM, Hong Kong China, 241–252.

[15] Hesham Elsayed, Mayra Donaji Barrera Machuca, Christian Schaarschmidt, Karola Marky, Florian Müller, Jan Riemann, Andrii Matviienko, Martin Schmitz, Martin Weigel, and Max Mühlhäuser. 2020. VRSketchPen: Unconstrained Haptic Assistance for Sketching in Virtual 3D Environments. In *26th ACM Symposium on Virtual Reality Software and Technology*. ACM, Virtual Event Canada, 1–11.

[16] Julien Epps, Serge Lichman, and Mike Wu. 2006. A study of hand shape use in tabletop gesture interaction. In *CHI'06 extended abstracts on human factors in computing systems*. ACM, Montréal Québec, Canada, 748–753.

[17] Mark Fuge, Mehmet Ersin Yumer, Gunay Orbay, and Levent Burak Kara. 2012. Conceptual design and modification of freeform surfaces using dual shape representations in augmented reality environments. *Computer-Aided Design* 44, 10 (2012), 1020–1032.

[18] Liuhao Ge, Yujun Cai, Junwu Weng, and Junsong Yuan. 2018. Hand pointnet: 3d hand pose estimation using point sets. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, Salt Lake City, UT, USA, 8417–8426.

[19] Yves Guiard. 1987. Asymmetric division of labor in human skilled bimanual action: The kinematic chain as a model. *Journal of motor behavior* 19, 4 (1987), 486–517.

[20] Sean Gustafson, Christian Holz, and Patrick Baudisch. 2011. Imaginary phone: learning imaginary interfaces by transferring spatial memory from a familiar device. In *Proceedings of the 24th annual ACM symposium on User interface software and technology*. ACM, Santa Barbara California USA, 283–292.

[21] Kelly S. Hale and Kay M. Stanney. 2014. *Handbook of Virtual Environments: Design, Implementation, and Applications* (2nd ed.). CRC Press, Inc., Boca Raton, FL, USA.

[22] Ken Hinckley, Michel Pahud, Hrvoje Benko, Pourang Irani, François Guimbretière, Marcel Gavriliu, Xiang'Anthony' Chen, Fabrice Matulic, William Buxton, and Andrew Wilson. 2014. Sensing techniques for tablet+ stylus interaction. In *Proceedings of the 27th annual ACM symposium on User interface software and technology*. ACM, Honolulu Hawaii USA, 605–614.

[23] Ken Hinckley, Koji Yatani, Michel Pahud, Nicole Coddington, Jenny Rodenhouse, Andy Wilson, Hrvoje Benko, and Bill Buxton. 2010. Pen+ touch= new tools. In

Proceedings of the 23nd annual ACM symposium on User interface software and technology. ACM, New York NY USA, 27–36.

[24] Bret Jackson and Daniel F Keefe. 2016. Lift-off: Using reference imagery and freehand sketching to create 3d models in vr. *IEEE transactions on visualization and computer graphics* 22, 4 (2016), 1442–1451.

[25] Roy S Kalawsky. 1999. VRUSE—A computerised diagnostic tool: for usability evaluation of virtual/synthetic environment systems. *Applied ergonomics* 30, 1 (1999), 11–25.

[26] Shaun K Kane, Jeffrey P Bigham, and Jacob O Wobbrock. 2008. Slide rule: making mobile touch screens accessible to blind people using multi-touch interaction techniques. In *Proceedings of the 10th international ACM SIGACCESS conference on Computers and accessibility*. ACM, Halifax Nova Scotia Canada, 73–80.

[27] Daniel Keefe, Robert Zeleznik, and David Laidlaw. 2007. Drawing on air: Input techniques for controlled 3D line illustration. *IEEE Transactions on Visualization and Computer Graphics* 13, 5 (2007), 1067–1081.

[28] Daniel F Keefe, Daniel Acevedo Feliz, Tomer Moscovich, David H Laidlaw, and Joseph J LaViola Jr. 2001. CavePainting: a fully immersive 3D artistic medium and interactive experience. In *Proceedings of the 2001 symposium on Interactive 3D graphics*. Citeseer, ACM, Chapel Hill, NC, USA, 85–93.

[29] David Kim, Otmar Hilliges, Shahram Izadi, Alex D Butler, Jiawen Chen, Iason Oikonomidis, and Patrick Olivier. 2012. Digits: freehand 3D interactions anywhere using a wrist-worn gloveless sensor. In *Proceedings of the 25th annual ACM symposium on User interface software and technology*. ACM, Cambridge Massachusetts USA, 167–176.

[30] Yongkwan Kim, Sang-Gyun An, Joon Hyub Lee, and Seok-Hyung Bae. 2018. Agile 3D sketching with air scaffolding. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. ACM, Montréal, Canada, 238.

[31] Yongkwan Kim and Seok-Hyung Bae. 2016. SketchingWithHands: 3D sketching handheld products with first-person hand posture. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*. ACM, Tokyo Japan, 797–808.

[32] Kin Chung Kwan and Hongbo Fu. 2019. Mobi3DSketch: 3D Sketching in Mobile AR. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. ACM, Glasgow Scotland UK, 176.

[33] Mayra D Barrera Machuca, Wolfgang Stuerzlinger, and Paul Asente. 2019. Smart3DGuides: Making Unconstrained Immersive 3D Drawing More Accurate. In *25th ACM Symposium on Virtual Reality Software and Technology*. ACM, Parramatta NSW Australia, 1–13.

[34] Keigo Matsumoto, Yuki Ban, Takuji Narumi, Yohei Yanase, Tomohiro Tanikawa, and Michitaka Hirose. 2016. Unlimited corridor: redirected walking techniques using visuo haptic interaction. In *ACM SIGGRAPH 2016 Emerging Technologies*. ACM, Anaheim California USA, 1–2.

[35] Franziska Mueller, Micah Davis, Florian Bernard, Oleksandr Sotnychenko, Mickeal Verschoor, Miguel A Otaduy, Dan Casas, and Christian Theobalt. 2019. Real-time pose and shape reconstruction of two interacting hands with a single depth camera. *ACM Transactions on Graphics (TOG)* 38, 4 (2019), 1–13.

[36] Franziska Mueller, Dushyant Mehta, Oleksandr Sotnychenko, Srinath Sridhar, Dan Casas, and Christian Theobalt. 2017. Real-time hand tracking under occlusion from an egocentric rgb-d sensor. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*. IEEE, Venice, Italy, 1284–1293.

[37] Andrew Nealen. 2007. *Interfaces and Algorithms for the Creation, Modification, and Optimization of Surface Meshes*. Doctoral Thesis. Technische Universität Berlin, Fakultät IV - Elektrotechnik und Informatik, Berlin. https://doi.org/10.14279/depositonce-1734

[38] Huaishu Peng, Jimmy Briggs, Cheng-Yao Wang, Kevin Guo, Joseph Kider, Stefanie Mueller, Patrick Baudisch, and François Guimbretière. 2018. RoMA: Interactive fabrication with augmented reality and a robotic 3D printer. In *Proceedings of the 2018 CHI conference on human factors in computing systems*. ACM, Montreal QC Canada, 1–12.

[39] Anna Pereira, Tevis Miller, Yi-Min Huang, Dan Odell, and David Rempel. 2013. Holding a tablet computer with one hand: effect of tablet design features on biomechanics and subjective usability among users with small hands. *Ergonomics* 56, 9 (2013), 1363–1375.

[40] Ken Pfeuffer, Benedikt Mayer, Diako Mardanbegi, and Hans Gellersen. 2017. Gaze+ pinch interaction in virtual reality. In *Proceedings of the 5th Symposium on Spatial User Interaction*. ACM, Brighton United Kingdom, 99–108.

[41] Dilip K Prasad, Maylor KH Leung, Chai Quek, and Siu-Yeung Cho. 2012. A novel framework for making dominant point detection methods non-parametric. *Image and Vision Computing* 30, 11 (2012), 843–859.

[42] Chen Qian, Xiao Sun, Yichen Wei, Xiaoou Tang, and Jian Sun. 2014. Realtime and robust hand tracking from depth. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. IEEE, Columbus, OH, USA, 1106–1113.

[43] Karthik Ramani et al. 2015. A gesture-free geometric approach for mid-air expression of design intent in 3D virtual pottery. *Computer-Aided Design* 69 (2015), 11–24.

[44] Steven Schkolne, Michael Pruett, and Peter Schröder. 2001. Surface drawing: creating organic 3D shapes with the hand and tangible tools. In *Proceedings of the SIGCHI conference on Human factors in computing systems*. ACM, Seattle Washington USA, 261–268.

[45] Jotaro Shigeyama, Takeru Hashimoto, Shigeo Yoshida, Takuji Narumi, Tomohiro Tanikawa, and Michitaka Hirose. 2019. Transcalibur: A weight shifting virtual reality controller for 2d shape rendering based on computational perception model. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. ACM, Glasgow Scotland UK, 1–11.

[46] Mike Sinclair, Eyal Ofek, Mar Gonzalez-Franco, and Christian Holz. 2019. Capstancrunch: A haptic vr controller with user-supplied force feedback. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology*. ACM, New Orleans, Louisiana, USA, 815–829.

[47] Hemant Bhaskar Surale, Aakar Gupta, Mark Hancock, and Daniel Vogel. 2019. TabletinVR: Exploring the design space for using a multi-touch tablet in virtual reality. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. ACM, Glasgow Scotland UK, 1–13.

[48] Ying-Chao Tung, Chun-Yen Hsu, Han-Yu Wang, Silvia Chyou, Jhe-Wei Lin, Pei-Jung Wu, Andries Valstar, and Mike Y Chen. 2015. User-defined game input for smart glasses in public space. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. ACM, Seoul Republic of Korea, 3327–3336.

[49] Gerold Wesche and Marc Droske. 2000. Conceptual free-form styling on the responsive workbench. In *Proceedings of the ACM symposium on Virtual reality software and technology*. ACM, Seoul Korea, 83–91.

[50] Gerold Wesche and Hans-Peter Seidel. 2001. FreeDrawer: a free-form sketching system on the responsive workbench. In *Proceedings of the ACM symposium on Virtual reality software and technology*. ACM, Baniff Alberta Canada, 167–174.

[51] Zheer Xu, Pui Chung Wong, Jun Gong, Te-Yen Wu, Aditya Shekhar Nittala, Xiaojun Bi, Jürgen Steimle, Hongbo Fu, Kening Zhu, and Xing-Dong Yang. 2019. TipText: Eyes-Free Text Entry on a Fingertip Keyboard. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology*. ACM, New Orleans LA USA, 883–899.

[52] H. Ye, K. C. Kwan, and H. Fu. 2021. 3D Curve Creation on and around Physical Objects with Mobile AR. *IEEE Transactions on Visualization and Computer Graphics* 1 (2021), 1–1e. https://doi.org/10.1109/TVCG.2020.3049006

[53] Ya-Ting Yue, Xiaolong Zhang, Yongliang Yang, Gang Ren, Yi-King Choi, and Wenping Wang. 2017. Wiredraw: 3d wire sculpturing guided with mixed reality. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. ACM, Denver Colorado USA, 3693–3704.

[54] André Zenner and Antonio Krüger. 2019. Drag: on: A virtual reality controller providing haptic feedback based on drag and weight shift. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. ACM, Glasgow Scotland UK, 1–12.