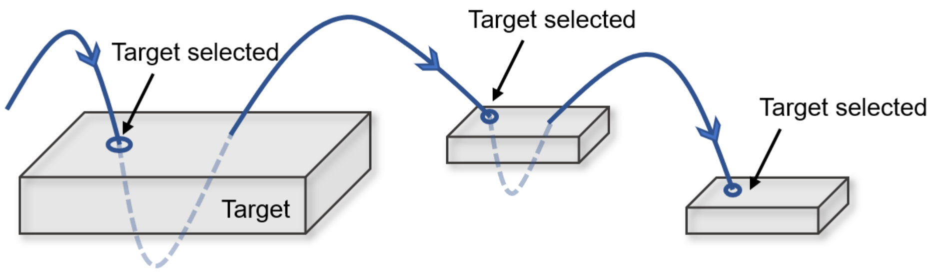
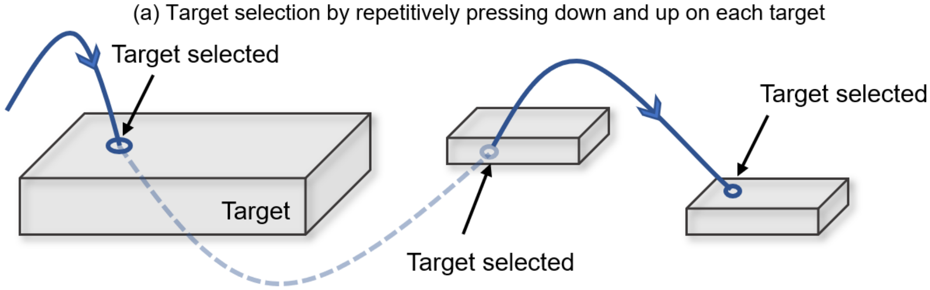
Graphical Abstract

Designing Successive Target Selection in Virtual Reality via Penetrating the Intangible Interface with Handheld Controllers

Yang Li, Sayan Sarcar, Kibum Kim, Huawei Tu, Xiangshi Ren



(a) Target selection by repetitively pressing down and up on each target



(b) Successive target selection by penetrating targets with one non-stop movement

Highlights

Designing Successive Target Selection in Virtual Reality via Penetrating the Intangible Interface with Handheld Controllers

Yang Li, Sayan Sarcar, Kibum Kim, Huawei Tu, Xiangshi Ren

- We discussed the penetrable feature of intangible displays and found the potential to facilitate target selection coherence among multiple targets.
- We proposed Sewing, a target selection method in VR for improving the coherence when users successively select multiple targets. With handheld controllers, users can finish selections by penetrating into one target and out directly towards another target.
- Investigations of Sewing showed that, with the real-life sewing metaphor on the fabric, users can learn and master the Sewing interaction quickly to finish VR target selection tasks.
- We also implemented SewTyping, a Sewing-based text entry method to facilitate users' VR typing performance. Evaluations revealed that SewTyping achieves promising typing speed (26.57 words per minute) and earned comments from participants as easy and engaging to use.

Designing Successive Target Selection in Virtual Reality via Penetrating the Intangible Interface with Handheld Controllers

Yang Li[a], Sayan Sarcar[b], Kibum Kim[c], Huawei Tu[d], Xiangshi Ren[a,*]

[a] *Center for Human-Engaged Computing, Kochi University of Technology, Kochi, Japan*
[b] *Faculty of Library, Information and Media Science, University of Tsukuba, Japan*
[c] *Department of HCI / Graduate School, Hanyang University, Seoul, Republic of Korea*
[d] *Department of Computer Science and Information Technology, La Trobe University, Melbourne, Australia*

Abstract

Empirical research on single target selection in Virtual Reality/Augmented Reality (VR/AR) environments has been studied extensively. However, there is a lack of methods and research on improving the coherence when successively selecting multiple targets. We propose Sewing, a successive target selection method with handheld controllers in VR environments. We leverage real-life sewing as a design metaphor for the selection of multiple targets with one single spatial movement penetrating targets with handheld controllers. We conducted an empirical study to validate the efficiency of selecting multiple targets with Sewing compared with the conventional selection method, which is based on the button pressing metaphor. Results showed that Sewing can provide promising performance and coherence for rapid and accurate target selection. Based on the results, we implemented SewTyping, a Sewing-based VR text entry method to evaluate the applicability of Sewing, and conducted an empirical study to compare the typing performance with two other techniques (Controller Pointing and Drum-like Keyboard). Results showed that 1) SewTyping achieved 26.57 words per minute with a total error rate of 3.68% and 2) users adapted to the sewing-like movement easily and achieved steady performance with essential practice.

Keywords: Target selection, Virtual reality, Text entry

*Corresponding author. Email address: ren.xiangshi@kochi-tech.ac.jp (Xiangshi Ren)

1. Introduction

Recent market trends show that Virtual Reality (VR) techniques have become popular and affordable among consumers [1]. VR also shows great capabilities and potential for applications in education [30], remote collaboration [35], and office work [20, 21]. Target selection is one of the fundamental operations [48, 54, 4, 25] that helps users acquire intended targets for further manipulations. Generally, users often need to manipulate multiple targets with a certain order to finish tasks such as selecting an item in a multi-layer menu or typing a sentence using the keys of a virtual keyboard. In such cases, there is a need for a target selection method that not only provides promising and reliable performance when selecting single targets but also contributes to a coherent selection experience when successively selecting multiple virtual targets.

Pointing [31] and crossing [2, 48] are two popular representative interaction paradigms which are currently applied in VR for target selection. Pointing allows users to select targets directly by poking (with fingers or controllers) or indirectly by navigating a cursor to locate the target and confirm it by approaches such as button pressing or dwelling. Crossing enables users to perform target selection by stroking through the boundary of the target. Previous research [48] validated the promising performance of crossing compared with pointing and mentioned the potential of crossing to achieve successive selections. However, we found that targets mentioned in most VR target selection techniques are not real 3D targets and they are only interactive from one side or in one direction (i.e., they lack the consideration of z-coordinate [10]). The interaction potentials of targets have therefore been dimensionally reduced to 2D or even 1D without fully leveraging the characteristics of 3D targets. Moreover, when selecting multiple targets successively, current methods based on pointing or crossing need to deal with targets individually and repetitively with less consideration of the interaction coherence among selections.

In this paper, we propose Sewing, a target selection method to facilitate the interaction coherence in VR when successively selecting multiple targets. Inspired by the real-life sewing metaphor with needles and fabric, we turn the penetrable feature [10] of intangible displays into an advantage to connect multiple targets into a single movement (as shown in Figure 1b). With the handheld controller and the virtual mallet (a virtual widget extended from the controller), multiple 3D virtual targets can be selected by penetrating the mallet down through from one of them and then moving directly to another target and penetrating up through it for the next selection. Sewing achieves coherent interaction performance by exploiting two sides (i.e., top and bottom sides) of the 3D targets, which is different from the previous pointing and crossing applied in current VR scenarios.

We first conducted an empirical study to evaluate the performance of Sewing and its potential to improve the coherence when users successively select multiple targets. Results showed that Sewing (alternatively penetrating the top and bottom side of targets for selection) provides a more accurate approach to select virtual targets successively with less time and spatial movement compared with the conventional method of bouncing among targets. Based on our findings, we applied Sewing in the context of VR text entry to explore its applicability and thus proposed SewTyping, a novel method to assist users to type faster and more fluently in VR environments. We further evaluated users' typing performance using SewTyping compared with Controller Pointing [41] and Drum-like Keyboard [7]. Results show that users achieved 26.57 (16.8 for Controller Pointing and 22.13 for Drum-like Keyboard) words per minute with an acceptable error rate of 3.68% using SewTyping (4.07% for Controller Pointing and 3.77% for Drum-like Keyboard). Participants also showed their preference for SewTyping and regarded it easy to learn and engaging to use. Beyond text entry, we also discussed the potentials and possibilities of leveraging Sewing into more scenarios such as spatial menu design and practical surgical training in VR.

2. Related Work

Target selection is the fundamental operation [4] that helps users to acquire target(s) for further manipulations. Based on the pointing and crossing paradigm [48, 2], numerous techniques have been proposed to enhance target selection performance in VR applications and systems. In this section, we first review current target selection methods. Then, focusing on different types of multiple-target selection tasks, we discuss whether and how existing methods and techniques could help users to select multiple targets effectively in VR scenarios.

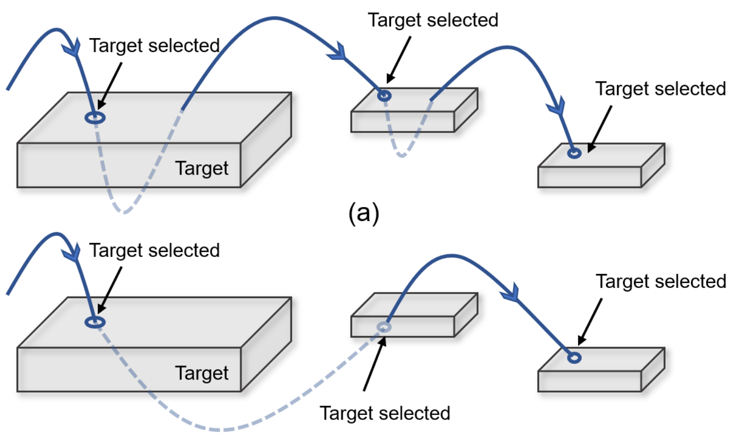


Figure 1: Two target selection methods: (a) Conventional up-and-down "pressing" (bouncing among targets) and (b) Sewing. When successively selecting targets, Sewing leverages the penetrable feature in VR to achieve the coherent target selection by penetrating targets as is done when sewing with a needle on the fabric.

2.1. Pointing-based Target Selection

Pointing is one of the vital ways for users to inform the interaction system about the intended target(s). To perform pointing in VR, users can leverage various tools such as hands [37, 33] (shown as avatars in VR scenarios), mice [45], handheld controllers [11], or styli [14]. Based on those tools, researchers further developed various methods to achieve satisfying performance. Currently, ray-casting is the popular and dominant component [4] in VR target selection designs. Furthermore, ray-based selection is the default choice in current commercial VR systems (e.g., HTC Vive and Oculus). In detail, users can irradiate the intended target by navigating a visible ray emitted from the physical tool (e.g., handheld controller) and confirm the selection with a trigger event [4, 12, 19, 42]. Ray-casting techniques show their advantage in operating targets far from the user's position [5] and relieving the physical load (e.g., walking towards distant targets). However, ray-casting techniques also face challenges when selecting small, dense, or occluded targets. Moreover, as the ray is emitted from the physical device, some behaviors (e.g., button pressing for confirmation) may cause subtle orientation changes which may affect pointing accuracy. Such changes will be enlarged by the length of the ray (distance to the target) and they will cause incorrect or missed selections [4, 8, 48].

2.2. Crossing-based Target Selection

The crossing paradigm was first introduced by Accot and Zhai [2] in 2002 that the target in 2D graphical user interfaces (GUI) can be selected by intersecting its boundary with the cursor. Cross Y [3] applies crossing-based selection in pen-based interaction and designs multiple crossing commands to trigger various functions. Dragicevic [15] combined the crossing and paper-based paradigm to simulate the real-life page-turning behavior to improve the drag-and-drop performance among overlapping windows. Wobbrock and Gajos [52] explored and validated the potential of crossing to assist people with motor impairments to select targets in user interfaces. Based on the crossing paradigm, various UI widgets were also proposed to help users set attributes [44] and manipulate multiple sliders [36]. For touch input, Luo and Vogel first validated the effectiveness of crossing compared with finger pointing [28] and then proposed Pin-and-Cross [29], which combines touch and crossing for unimanual menu operations.

Tu et al. [48] were the first to transplant the crossing paradigm to the VR target selection. They used ray-casting with handheld controllers to achieve boundary crossings of 2D and 1D targets in VR environments. Further validation showed that VR crossing can also be well modeled by Fitts' Law [17]. Instead of casting the ray from controllers, Yan et al. [55] navigated the pointer via head movements; users move the pointer into the target and back across the boundary to finish the selection. The above research made crossing paradigm available in the VR environments. Whereas, the targets for such availability were discussed with reduced dimension rather than 3D. It made us curious whether crossing or similar interaction can be implemented with real 3D targets.

2.3. Multiple-Target Selection Solutions in VR

To the best of our knowledge, the majority of research on multiple-target selection in VR focused on the large-scale selection [43] that the number of targets is counted in dozens or hundreds (e.g., joints of a 3D model or one part of the data point cloud) with one single selection. Lucas et al. [27] proposed Selection Box and Tablet Freehand Lasso to assist users to select multiple targets. Selection Box uses a semi-transparent volume selector as used in [59] to acquire targets touching or within the selector. Tablet Freehand Lasso allows users to draw a 2D arbitrary shape on a screen that shows the camera view. Then the camera projects a 3D area based on the shape to choose the targets within the projection volume. Stenholt [43] selected multiple targets simultaneously through a 3D-based spherical brush with adjustable radius, spatial lasso, or an algorithm based on the seed object and gestalt law of proximity [51] (the technique was named Magic Wand). Benavides et al. [6] allowed users to choose the intended pile of rendered spatial data points with bare-hand mid-air gestures and movements. Slicing-Volume [34] uses a virtual volume to roughly acquire the intended targets of a dense data cloud and then to finish the detailed selection on a tablet with handheld controllers and a stylus.

In real-life interaction scenarios, users often need to serially interact with multiple targets (e.g., selecting an item from a multi-layer menu or typing on keyboards to select characters) in VR environments. However, research on improving selection performance in those conditions is lacking. To the best of our knowledge, only Tu et al. [48] mentioned the potential of VR crossing for successive selections. However, VR crossing treated targets (e.g., virtual plates) as 2D or 1D representatives rather than real 3D ones considering targets' three dimensions.

Theoretically, single target selection techniques can be used to acquire multiple targets by repetitive execution of the selection procedure [27, 43]. However, single target selection techniques cannot achieve coherent performance in multiple-target selection tasks for the following reasons. First, although intended targets are interrelated from the task level, users need to handle every target independently and repetitively based on the same and repetitive operation process with less consideration about effective transitions among targets. Second, based on the optimized initial impulse model [32], when navigating the selector towards the intended target (except for discrete methods such as pressing arrow keys), users generally perform a fast but inaccurate movement to roughly travel towards the target and then slow down to reach the target for further manipulations (especially for small targets). Frequent velocity changes may not foster a sense of continuity and coherence when serially selecting multiple targets. Third, some target selection techniques contain additional mechanisms such as button click [19], dwell [53], or gestures [23, 55] after selector navigation to confirm selections. Thus, during selections, users would encounter frequent and multiple times of mode switching, which may influence the smoothness among selections and bring extra time consumption.

3. Sewing: a Successive Target Selection Method

In this section, we illustrate how Sewing works for multiple target selection and discuss similarities and differences between crossing [48] and Sewing.

3.1. Interaction Design

From real-life sewing with needles on fabrics, we derived an interaction method that allows users to select multiple targets successively in VR environments with smooth and coherent movement. Here we describe the design details of Sewing.

Learning from the Drum-like Keyboard [7], we extend a virtual mallet (with a small sphere on the tip) from the top of the controller because it is difficult to select the target (especially due to the limited size) using the whole body of the handheld controller. Users move the handheld controller and use the virtual mallet to perform target penetration.

Then, to simulate the real-life sewing movement in VR, we make all virtual targets interactive from both the top and the bottom side of the virtual targets. Users navigate the handheld controller towards the virtual target and finish the selection by hitting either the top or the bottom side of the target. Selections are confirmed as the penetration occurs. Theoretically, penetration can happen on any side of 3D targets, such as left or right side of a cuboid object. As an initial exploration of Sewing, in this paper, we only consider the penetrations that happens on the top and bottom side.

Double-side interactive targets and the handheld controller (with virtual mallets) make it possible to perform sewing-like movements. After hitting one target, there is no need for a backward movement (see the dashed part in Figure 1a). Instead, as shown in Figure 1b, users can move the controller directly to the following target and finish the selection. In this case, all targets will be "sewn" together with a single and coherent movement rather than repetitive up-and-down movements.

3.2. Similarities and Differences

From the interaction perspective, Sewing shares similarities with crossing-based [2] target selection in VR [48]. First, they both perform a movement starting from one side to another side of the object. Second, selection for both methods can be immediately confirmed when penetrating or crossing the boundary of the target. Third, both of them consider continuous movement as a solution for successive selections.

However, there are also differences between these two methods. First of all, Sewing uses a virtual mallet (rather like a sewing needle) to simulate the sewing movement (i.e., penetrating the surface of a 3D virtual object). Crossing is the operation that behaves more like cutting an object with a beam (i.e., crossing the boundary of the object). Second, targets for Sewing are 3D ones with two (or more) interactive surfaces which allow users to alternatively penetrate targets from either the top side or the bottom side of the target permitting smooth and coherent transitions between targets; by contrast, crossing focuses on targets that have only one interactive side due to the decreased number of active dimensions (e.g., 2D targets in 3D environments, and 1D targets in 2D environments). Third, Sewing focuses on the enhancement of coherence when selecting multiple targets, while crossing focuses more on single target selection with less consideration of transitions between targets.

4. Empirical Study

We conducted an empirical study to validate the feasibility of Sewing when successively selecting multiple targets in the VR environment. In this study, we investigated the influence of various factors on target selection performance. Based on the results of this study, we validated the following hypotheses:

a. Sewing is able to reduce target selection time and spatial movement compared with the conventional method.
b. Sewing is more accurate than the conventional method when selecting targets successively.

4.1. Participants and Apparatus

12 participants (10 males and 2 females, aged between 21 and 24 years old, M = 22.58, SD = 1.08, all right-handed) volunteered in this study. None of them had experience using head-mounted displays (HMDs) and hand-held controllers before the study. All participants had normal vision (or corrected-to-normal vision by wearing glasses). The experiment system was implemented with Unity 5.6, HTC Vive, and its handheld controllers. The system ran on a desktop computer with Intel i7-3770 CPU, 16 GB RAM, NVIDIA Quadro K4000 graphics card, and Windows 10 operating system.

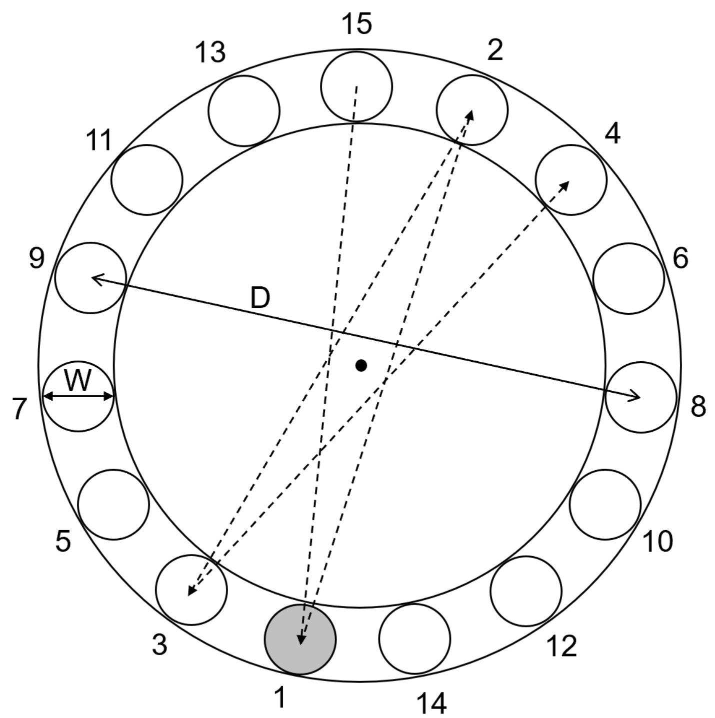


Figure 2: 2D-perspective schematic layout for the target selection task based on ISO 9241-9. It should be noted that all targets are 3D ones in our study. W means target size, D stands for the distance between two targets with adjacent numbers. The experiment begins by selecting the top target (number 15), then participants follow the path shown with dotted arrow lines to select targets successively. The next intended target is highlighted (in green) after the previous target has been selected.

4.2. Experimental Design

We designed the target selection task based on the ISO 9241-9 standard [22], which was commonly used in previous research [38, 40, 46, 58]. 15 circular 3D targets were arranged in an annular layout (see Figure 2 as the schematic). The experiment took the following three factors into consideration: selection method, index of difficulty (ID, which is related to the target size and target distance), and the orientation of the target.

Two target selection methods we evaluated were: Bouncing and Sewing. Bouncing allows users to select the target only by hitting the top side of the target with the virtual mallet. As the mallet would travel in or through the target, users need to move the mallet back above the target (homing action) after the hitting. By contrast, selection using Sewing can occur when the mallet penetrates from either the top or the bottom side of the target. Then, the mallet can pass directly through the target and on to the next target, i.e., no homing action is required after the penetration in Sewing.

Target size was defined as the diameter of targets; target distance was calculated as the length between two target centers (see Figure 2). To evaluate the target selection speed and accuracy under various target sizes and distances, we designed a combination of 4 target sizes (5mm, 10mm, 20mm, 40mm in radius) $\times$ 2 target distances (320mm, 640mm). We calculated the index of difficulty (ID) for all size-distance combinations. IDs covered the range between 2.32 and 6.02. We used the ID as the independent variable (instead of target size and target distance) for the following analysis because 1) ID is a competent factor for describing the selection difficulty of the intended targets in VR target selection studies [48, 38], and, 2) using the ID can decrease the number of independent variables, which would improve the interpretability of the results.

Tilt degree described the gradient condition of the panel of virtual targets against the ground rendered in the VR environment. Here we set two tilt degree options, vertical and horizontal, in this metric for the following experiment.

4.3. Task and Procedure

The experiment lasted around 45 minutes for each participant. We first introduced the experiment to participants and guided them to stand at the center of the experiment area with the HMD on the head and a handheld controller in the dominant hand. We provided enough time for all participants to get used to the VR system and to practise two mentioned methods with try-out selections. The height of all targets and the distance between participants and targets were adjusted during the practice (participants can move their standing position slightly within a circle of 20cm radius). Then, participants were requested to select targets with the assigned selection method using the virtual mallet as fast and accurate as possible. At the beginning of the experiment, the target on the top (see Figure 2) turned red. Participants needed to select the right target to trigger ensuing selections. After selecting the red target, the red target turned white and the next target turned green. During the selection, it should be noted that the next target would not turn green until the previous target was selected successfully. The sequence of selections followed the labeled index numbers in Figure 2.

The order of using two selection methods was counter-balanced across all participants. For every participant, 16 sets of combinations (target size × target distance × tilt degree) were conducted in random order. Each set contained 15 target selections. In summary, the number of target selections was: 12 participants × 2 selection methods × 4 target sizes × 2 target distances × 2 tilt degrees × 15 target selections = 5760.

4.4. Results and Analysis

We removed outliers (4.76%) outside the 95% confidence interval and used the remaining 95.24% of the dataset for the following analysis. As the data did not pass the normality check, we analyzed the data with the Generalized Linear Mixed Model (GLMM via the R package named 'lme4'). Table 1 summarizes our main statistical findings from the perspective of selection time, interaction depth, travel distance, and selection error rate.

4.4.1. Selection Time

Selection time is defined as the duration between the selection of the previous target and the moment when participants select the following green target. In general (see Table 1 for details), Sewing required less time than Bouncing when selecting targets. It required less time to select targets vertical to the ground than those horizontal to the ground. With the increase of ID, participants needed more time to select targets. GLMM analysis showed that selection method ($\chi^2(2) = 9691.9$, $p < .001$), tilt degree ($\chi^2(2) = 62.8$, $p < .001$), and ID ($\chi^2(5) = 30425$, $p < .001$) had a significant main effect on selection time. The interaction effect of selection method × tilt degree ($\chi^2(4) = 11.36$, $p < .001$), selection method × ID ($\chi^2(10) = 2186.1$, $p < .001$), and tilt degree × ID ($\chi^2(10) = 37.25$, $p < .001$) was also significant.

4.4.2. Interaction Depth

Interaction depth signifies the distance between the furthest position of the path of the virtual mallet and the target plane when selecting the target. Sewing required less interaction depth than Bouncing when the virtual mallet traveled among targets. When targets were positioned vertical to the ground, less interaction depth was used than the horizontal condition. GLMM analysis showed that selection method ($\chi^2(2) = 271.86$, $p < .001$), tilt degree ($\chi^2(2) = 84.72$, $p < .001$), and ID ($\chi^2(5) = 65.68$, $p < .001$) had a significant main effect on interaction depth. We also found a significant interaction effect of selection method × ID ($\chi^2(10) = 13.15$, $p < .001$), and tilt degree × ID ($\chi^2(10) = 4.2$, $p < .05$).

Independent Variables and Candidates		Selection Time/ms	Interaction Depth/cm	Travel Distance/cm	Error Rate/%
Selection Method	Bouncing	977.72 (210.66)	17.18 (5.86)	78.35 (21.92)	3.94 (2.28)
	Sewing	750.70 (180.15)	9.37 (3.32)	58.41 (17.65)	1.35 (1.24)
Tilt Degree	Vertical	837.35 (214.47)	10.91 (4.99)	66.13 (21.89)	2.46 (1.41)
	Horizontal	891.13 (234.97)	15.65 (6.3)	70.63 (22.4)	2.82 (1.73)
ID	2.32	657.05 (141.62)	11.64 (5.09)	50.69 (11.8)	0.84 (1.51)
	3.17	789.18 (209.48)	13.7 (6.4)	69.4 (23.18)	2.34 (2.04)
	4.09	856.98 (187.31)	13.07 (6)	68.14 (21.79)	2.77 (1.47)
	5.04	942.21 (193.17)	12.98 (6.32)	68.63 (22.39)	3.82 (2.68)
	6.02	1123.59 (169.42)	15.31 (6.07)	86.47 (12.5)	2.6 (2.89)

Table 1: Overview of the statistical results for target selection performance in various scales: selection method, tilt degree, and ID. All results in the Table referred to mean values and numbers in brackets refer to standard deviations. All independent variables showed the significant (all $p < .001$) effect on all dependent variables (columns) except Tilt Degree on Selection Error ($p = 0.37$).

4.4.3. Travel Distance

Travel distance refers to the spatial trace length of the virtual mallet in the virtual space during the selection. Results show that Sewing needed less travel distance to finish the selection. Selecting targets vertical to the ground also required less travel distance. With increases in the ID, more travel distance was needed to select targets. GLMM analysis showed that selection method ($\chi^2(2) = 379.28$, $p < .001$), tilt degree ($\chi^2(2) = 6.95$, $p < .001$), and ID ($\chi^2(5) = 1130.15$, $p < .001$) had a significant main effect on travel distance. The interaction effect of selection method $\times$ tilt degree ($\chi^2(4) = 13.4$, $p < .001$) and selection method $\times$ ID ($\chi^2(10) = 23.2$, $p < .001$)

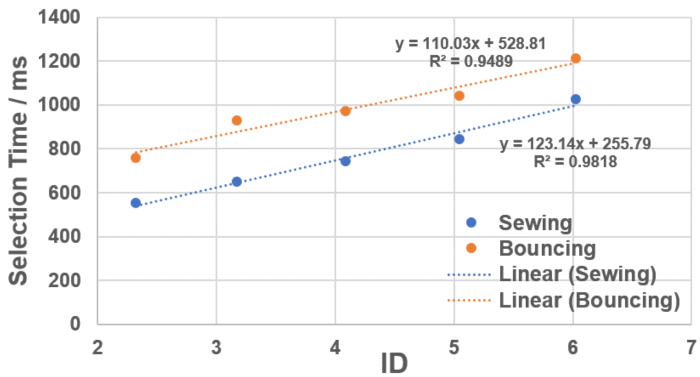


Figure 3: Fitts' Law modeling for the conventional Bouncing and Sewing interaction.

was also significant.

4.4.4. Selection Error

We define a selection error as a trial that participants did not successfully select the target on the first attempt. Selection error rate denotes the percentage of trials that participants made selection errors. In general, participants made fewer selection errors with Sewing than Bouncing. With the increase of ID, selection error rate also increased and reached the peak of 3.82% when ID was 5.04. GLMM analysis showed that selection method ($\chi^2(2) = 10.63$, p <.001) and ID ($\chi^2(5) = 15.45$, p <.001) had a significant main effect on selection error. Although participants had less chance of making selection errors for targets that were vertical to the ground, no significant main effect was founded for the tilt degree (p = 0.37). Results also revealed a significant interaction effect of selection method $\times$ ID ($\chi^2(10) = 8.05$, p <.01).

4.4.5. Fitts' Law Modeling

The data of Sewing was validated as a good fit by Fitt's Law as shown in Figure 3 (with the R-square of 0.98). This result inferred that the interaction of Sewing can be well-modeled and described by the Fitts' Law. The regression line of Sewing and Bouncing would intersect at a high ID of 20.83, which indicated that Sewing could serve as an alternative selection method of the conventional bouncing when successively selecting multiple targets in VR environments.

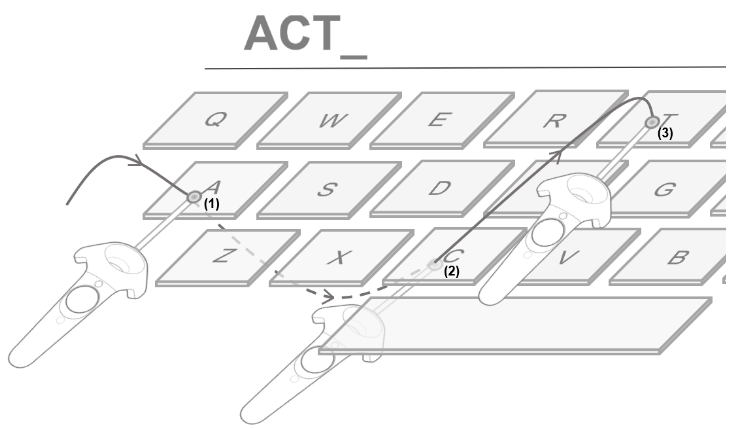


Figure 4: Text entry with SewTyping in VR. Characters (3D buttons shown in the figure) are successively selected with the virtual mallet extended from the handheld controller by penetrating on either the top or the bottom side of virtual keys. E.g., to enter the word “ACT”, “A” and “T” are selected from the top side, while “C” is selected from the bottom side.

5. Sewing-based Typing in Virtual Reality

After validating the performance of Sewing, we applied Sewing in the context of VR text entry and implemented a novel technique, named SewTyping, to facilitate users’ typing efficiency in VR scenarios. We chose text entry as the application scenario for Sewing because text entry is a representative task that requires users to successively select multiple targets (characters) to generate and enter text into computer systems. We further conducted a comparative study to evaluate users’ typing performance using SewTyping compared with Controller Pointing [41] and Drum-like Keyboard [7].

Based on the Sewing method, SewTyping renders the virtual keyboard double-side interactive. Users can use the handheld controller (with the virtual mallet on the top) to perform sewing-like movements among character keys. For instance, users enter the word “ACT” (shown in Figure 4) by penetrating down through “A”, up through “C”, down through “T”. For words with repetitive characters (e.g., “WOOD”), users can penetrate through “W”, up through “O”, down through “O”, and up through “D”. For the use of backspace, one character can be deleted when the backspace key is penetrated from either the top or bottom side.

5.1. Participants and Apparatus

We recruited 27 participants for the experiment (11 males and 16 females, aged between 22 and 46 years, M = 26.56, SD = 5.55, all right-handed, none of them being native English speakers). All participants had normal vision without color blindness (19 of them wore glasses for short-sighted correction). None of them had VR text entry experience before the experiment. All participants were able to comprehend English sentences and were familiar with the QWERTY keyboard according to their self-report.

The experiment system was implemented based on the Cutie keys [13] in Unity 5.6 with HTC Vive and its handheld controllers. We set up a spatial area of 2m (length) $\times$ 2m (width) $\times$ 2m (height) tracking space with two HTC Vive optical trackers. We used a virtual keyboard with a QWERTY layout. We also ran a pilot study and confirmed the key size 10cm $\times$ 10cm, a 3cm gap between keys (We did not use additional visual feedback, such as highlight or projections, when the mallet was occluded by keys as it would bring distractions to participants during text entry), 32cm length for the virtual mallet, 1.5cm radius for the sphere on the top of the virtual mallet forming a balance between the detection quality of the system and the participants' typing experience. Intelligent-aid techniques such as auto-correction and auto-implementation functions were disabled to avoid the distraction of incorrect or unexpected auto-entry conditions. The desktop used in this study is the same as that used in the previous Sewing empirical study.

5.2. Task and Corpus

In the experiment, all participants were requested to transcribe sentences with two controllers in the experiment system using the assigned technique as fast and accurate as possible. If any typo was detected during the text entry process, participants had to correct it with the backspace on the virtual keyboard.

We used a subset of the Enron Mobile Email Dataset [50] (107 sentences, with 20-40 characters each, all in lowercase without any punctuations or numbers) as the corpus because: 1) all sentences are used on a daily basis and they are easy to memorize [26] and, 2) the corpus has been used to evaluate VR text entry techniques in the previous research [41]. Additionally, using sentences (rather than phrases) can also simulate real text entry scenarios (e.g., sending sentences in a VR messaging system).

5.3. Design and Procedure

According to the literature, we chose Controller Pointing and Drum-like Keyboard as the comparative techniques for VR text entry with virtual keyboards and handheld controllers. The reasons for choosing these two techniques are: 1) Controller Pointing uses pointing (with a beam) as the selection method, which is currently one of the most fundamental patterns [24, 41]; Drum-like Keyboard uses the selection method that simulates the button pressing on a physical keyboard and, 2) both of these two techniques show promising text entry speed and error rates as reported in the literature [7, 41] (in the context of VR typing with virtual keyboards). In order to be consistent with the evaluation designs as conducted in [41] and [7], we asked participants to finish the text entry task with two handheld controllers in their hands. Here we describe the operational procedures of the three evaluated VR text entry techniques:

- *Controller Pointing:* participants navigate the ray emitted from the handheld controller to point at the intended key and confirm the selection by pressing the trigger button on the controller.
- *Drum-like Keyboard:* participants use the virtual mallet extended from the head of the handheld controller to select the character by hitting the character button which is effective from the top side only.
- *SewTyping:* participants use the virtual mallet extended from the head of the handheld controller to penetrate down or up through buttons to select the intended characters (as shown in Figure 4).

The ray emitted from the handheld controller (for Controller Pointing) and the virtual mallet (for Drum-like Keyboard and SewTyping) were visible during the experiment. When the button was irradiated by the ray or hit by the virtual mallet, it was highlighted in light red. Haptic feedback was provided as the participant pressed the trigger of the handheld controller (for Controller Pointing) or as the button was hit (for Drum-like Keyboard and SewTyping).

To avoid potential interference on data quality from the various operation methods of the three techniques (especially for Drum-like Keyboard and SewTyping), we chose a between-subjects design for the experiment. The independent variable was the technique. We randomly and evenly arranged 27 participants into three groups. For each group, participants were instructed

Figure 5: The setup for the SewTyping experiment. The participant (with two controllers in his hand) was seated on a chair in the middle of the space which was detected by optical trackers.

with only one text entry technique and they were requested to transcribe 50 sentences (in five sessions, similar to the design in [57]) randomly chosen from the corpus. In total, the number of transcriptions was: 3 techniques $\times$ 9 participants per technique $\times$ 50 sentences = 1350.

We first informed participants about the purpose of this study. After that, we gave a tutorial on the text entry technique assigned to them (due to the between-subjects design). Then we instructed and assisted participants to wear the HMD, to hold controllers with their left and right hands, and to sit on the chair in the center of the experiment area where their movement could be detected by the HTC Vive optical trackers (see Figure 5). Participants had enough time (around 7-10 minutes) to practise the assigned technique before the formal experiment. During the practice, we also helped participants to adjust the height and orientation towards the virtual keyboard (participants can slightly adjust their position within 20cm radius to the center of the experiment area ground and height within 10cm).

The target sentence first appeared on the experiment interface. Then participants were required to transcribe the sentence and submit the input by pressing the button on the handheld controller. Participants could have a 2-minute rest after every 10 sentences.

After the experiment, we used the questionnaire deployed in [57] (five-scale, one for bad, five for good) to collect participants' perceived ratings on speed, accuracy, fluidity (interaction coherence), fatigue, learnability, and preference. System Usability Scale (SUS) [9] was also used to collect participants' subjective feedback and comments.

5.4. Results

We filtered the data with 3 times the standard deviation of the backspace number and removed 30 (2.22%) outliers. As the data didn't pass the normality check, we analyzed the data with the Kruskal-Wallis test and the post-hoc test using Mann-Whitney tests with Bonferroni correction.

5.4.1. Words per Minute (WPM)

For every transcribed sentence, the typing speed was calculated as the number of words (every five characters) divided by the transcription time (in minutes) [16]. Overall, SewTyping got the highest WPM (M = 26.57, SD = 5.66). Drum-like Keyboard got a higher WPM (M = 22.13, SD = 4.49) than Controller Pointing (M = 16.8, SD = 4.29). A Kruskal-Wallis test revealed a significant effect of technique on the typing speed ($\chi^2(2) = 538.97$, $p < .001$). The post-hoc testing using Mann-Whitney tests with Bonferroni correction showed significant differences among the three techniques (all $p < .001$). Figure 6 shows the typing speed of the three techniques among five sessions.

5.4.2. Inter-key Interval (IKI)

Inter-key interval (IKI) is the duration (in milliseconds) between two button selections [47] (including space and backspace key). We used IKI to evaluate the coherence of text entry. SewTyping achieved the shortest IKI (M = 431.94, SD = 80). Drum-like Keyboard got less IKI (M = 514.77, SD = 95.81) than Controller Pointing (M = 686.97, SD = 157.89). A Kruskal-Wallis test revealed a significant effect of technique on the IKI ($\chi^2(2) = 655.63$, $p < .001$). Post-hoc testing using Mann-Whitney tests with Bonferroni correction showed significant differences among the three techniques (all $p < .001$). Figure 7 shows the IKI of three techniques among five sessions.

5.4.3. Total Error Rate

As the experiment forced participants to correct typos before submitting their transcriptions, we only evaluated the total error rate (equal to the cor-

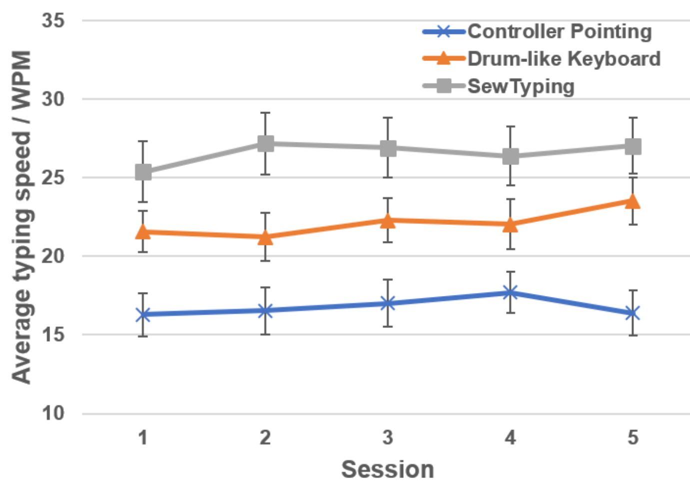


Figure 6: The average typing speed for the three techniques in five sessions. Error bars represent the standard error.

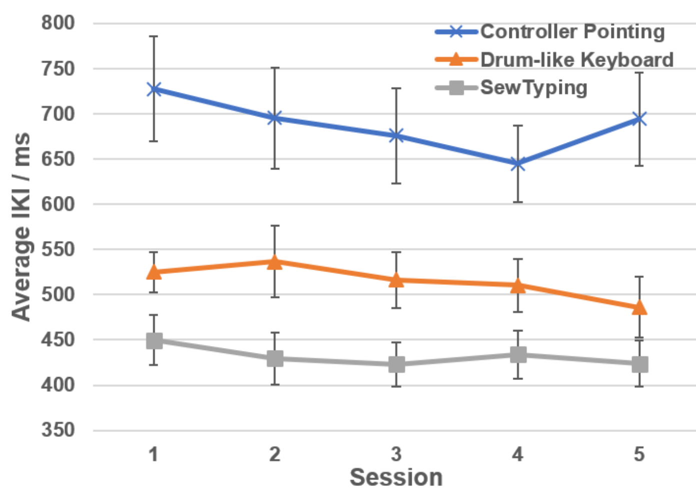


Figure 7: The average Inter-key Interval (IKI) for three techniques in five sessions. Error bars represent the standard error.

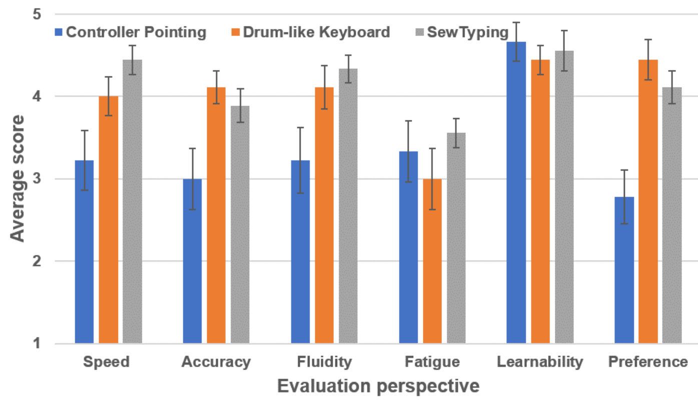


Figure 8: Participants' feedback regarding six evaluation perspectives (1 for bad, 5 for good). Error bars represent the standard error.

rected error rate). Total error rate was calculated as the number of incorrect-but-corrected characters divided by the sum of correct and incorrect-but-corrected characters [39]. The order of the average total error rate for the three techniques was: Controller Pointing (4.07%) >Drum-like Keyboard (3.77%) >SewTyping (3.68%). A Kruskal-Wallis test did not find a significant difference of technique on the total error rate (p = 0.94).

5.4.4. Subjective Evaluations

SUS results showed that Controller Pointing got the lowest score (M = 70.6, SD = 9.58, calculated by the SUS protocol [9]), Drum-like Keyboard with an average of 75.8 (SD = 12.25), and SewTyping with an average of 75.8 (SD = 7.4). All three techniques got high ratings (the higher the better) on question No.7: "I would imagine that most people would learn to use this system very quickly").

Evaluations according to the participants' subjective perceptions were obtained via a 5-scale questionnaire (see Figure 8). Results showed that SewTyping achieved the highest scores on speed (M = 4.44, SD = 0.18), fluidity (M = 4.33, SD = 0.17), and fatigue (M = 3.56, SD = 0.18). A Kruskal-Wallis test revealed a significant effect of technique on speed ($\chi^2(2)$ = 6.1, p <.05), accuracy ($\chi^2(2)$ = 6.41, p <.05), and preference ($\chi^2(2)$ = 10.91, p <.01).

6. Discussion

Here we discuss interesting findings from the studies we conducted and further explore the potential of applying Sewing into other VR application scenarios.

6.1. Sewing: the Target Selection Method

Results of the study indicated that, compared with the conventional Bouncing method, Sewing was faster and more accurate with less spatial movement when participants successively selecting multiple targets in the VR environment (the promising results also validated our hypotheses). Sewing led to less time consumption and travel distance for the following reasons: 1) Sewing provided an integrated perspective to consider all selection targets as joints in a coherent and flexible movement instead of individual units which should be treated separately (with repeated manipulation). 2) Sewing made each movement and each penetration during the selection productive and meaningful, positively contributing to the selection of the following character(s). After penetrating the target, participants were able to move forward towards the following target without re-positioning the virtual mallet above the target panel (thus further decreasing the time consumption and spatial movement). 3) Less interaction depth also indicated that the track for Sewing is more direct between and among targets without large fluctuations. SewTyping's double-side interactive targets also provided the potential for quick re-selection if the participant failed to select the target at the first penetration attempt (e.g., penetrated the target panel but out of the target contour).

Results also indicated that participants performed better (in terms of selection time, interaction depth, and travel distance) when targets were positioned vertically to the ground. This finding may serve as supportive design rationales in existing studies about locating their interaction panels at certain angles [54] or vertical to the ground [41, 57]. The possible reason for getting this result may be that, due to the limited visual angle of the HMD and the position of targets, participants needed to move back and forth if the targets are hard to reach or out of sight. Especially for horizontally-positioned targets that were quite near to participants, they needed to step back a little while moving the virtual mallet, which may cause the longer spatial movement and time. Additionally, when selecting horizontally-positioned targets, participants needed to maintain the posture of putting their heads down with the heavy HMD. The extra physical load and fatigue may further influence participants' performance when selecting horizontal targets.

6.2. SewTyping

Compared with Controller Pointing and Drum-like Keyboard, SewTyping produced the shortest IKI, which implies that SewTyping could reduce the duration of key selections and enhance the coherence of text entry. The reason is that, with participants' familiarity with the keyboard layout, we could leverage the momentum of the penetration movement and save the time and movement by allowing them to move directly through to the next key for subsequent selections (based on the double-side interactive keyboard). Controller Pointing showed longer IKI for character selection because it requires time to complete the confirmation (by pressing the trigger button), to release the trigger button after confirmation, and to search for the next key. Meanwhile, the movement of pressing the trigger may cause cascade shift of the beam, which may increase the time for participants to enter a character with Controller Pointing.

From the design perspective, SewTyping avoids repetitive up-and-down character selection movements and sudden changes in acceleration, which may cause neuromuscular fatigue [18]. Although Drum-like Keyboard produced increasing typing speed as the session proceeded, participants also encountered extra fatigue during the speed-up. One participant using Drum-like Keyboard commented that: *"I usually strike down keys quickly and hard, thus it brings more fatigue (and it consumes more effort) to stop the strike and start the next one, especially when I attempted to improve my typing speed"*. During the practice, it was interesting to observe that participants developed a similar strategy to mitigate the fatigue brought from the technique assigned to the participant. All participants chose to keep elbows close to their bodies, mainly moving their forearms during the experiment. This strategy reduced the chance of moving the whole arm when using the assigned technique.

The high learnability rating inferred that SewTyping is easy for participants to use. There are two reasons for this. First, SewTyping maintains the QWERTY layout, which mitigates the burden of learning a new keyboard. Second, the familiarity inspired by the sewing metaphor helps participants understand the way to interact and adapt to the new technique. Additionally, the sewing metaphor changes the VR text entry from a task to engaging virtual sewing gameplay with needles (handheld controllers) on the fabric (the intangible virtual keyboard).

When typing with two hands, SewTyping also enables the sewing-like movement to enter characters fluently with two hand-held controllers interchangeably penetrate the character buttons. The possible reason for such interaction is that, participants may further leverage the previous experience of allocating keys to different hands/fingers when using other devices such as physical/virtual keyboards. When the character is penetrated and the following one is not proper for the current hand to perform the penetration, participants may be able to aware this situation subconsciously and use the proper strategy (i.e., another controller) to continue the penetration. This means that, with such experience, participants can avoid long-distance stretching (e.g., from "z" to "o") by dynamically using the capable controller (to penetrate the intended character) and thus improve the typing speed and coherence.

SewTyping enters text with a flexible input unit. Sewing can arrange characters (with any length), even the text revision process (i.e., the use of backspace) into a single and coherent movement. Compared with character-level techniques (e.g., Controller Pointing and Drum-like Keyboard), SewTyping enhances the coherence of text entry with fewer pauses between characters. As for word-level techniques (e.g., GestureType from [57]), SewTyping allows revision while participants are observing typos or improper content (e.g., a misused word) rather than after performing a word-level gesture. It is surprising that SewTyping achieved higher typing speed (26.57 WPM) than GestureType (24.73 WPM, as reported in [57]) without the risk of dizziness caused by the frequent head-shaking movement.

According to subjective feedback regarding the provided text entry technique, participants who use Drum-like Keyboard and SewTyping gave higher scores on the perspective of preference (than participants with Controller Pointing). The possible reason are: 1) Drum-like Keyboard and SewTyping leverage and simulate real-life behaviors. Life experience of participants reduce the learning cost and make those techniques easy to understand and 2) game-like experience of those two techniques may make it more engaging to enter text as a game rather than just a task. For SewTyping, it also brings surprising experience to participants as it turns the penetrable feature of intangible displays into the advantage of coherent typing by penetrating up and down among keys, which cannot happen on physical keyboards.

6.3. More Potential Application for Sewing

In this paper, we mainly implemented Sewing for facilitating VR text entry. Besides that, we also found potentials to apply the Sewing metaphor in more scenarios:

- *Spatial gesture design:* Sewing sheds light on extending input channels (e.g., spatial orientation or amplitude when performing the gesture) for current gesture designs. For instance, in a VR wizard game, users can perform a vertical "N" gesture (with the handheld controller) for the magic spell with normal power. They can also perform a horizontal larger "N" gesture for the same spell, but with enhanced power to strike the enemy.

- *Spatial menu design:* 1) We can implement a spatial multi-layer menu and use Sewing to realize the coherent process of calling out the menu, selecting the category, and selecting the intended item under the selected category with a single controller movement. 2) We can also arrange various functions into categories and integrate them into only one virtual button. Users can trigger different categories by penetrating the button at different angles and then penetrate the intended item for selection.

- *Non-visual text entry:* Similar to [56], with trajectory detection and machine learning algorithms, Sewing and users' familiarity with the QWERTY layout can be leveraged to achieve freehand text entry in VR without showing or paying visual attention to the virtual keyboard.

- *Surgical training:* VR applications can be implemented based on Sewing to stimulate stitching operations and penetration for medical training (e.g., surgical wound closure and intravenous injection).

7. Limitation and Future Work

As a novel target selection method, we initially validated the feasibility of Sewing in successively selecting 3D targets in VR and used text entry as an instance to leverage and evaluate the performance of Sewing interaction. Meanwhile, there needs more research to fully explore and understand the potential of Sewing.

The first limitation is that, in this paper, we only enable Sewing interaction with two (i.e., top and bottom) sides of 3D targets. If the thickness (the distance between two sides) of the target is enough, it would allow more target sides that can be penetrated. In that case, combinations of penetrated (in and out) sides and possible gestures during penetration (inside the 3D target) can enrich the interactivity of a single object. Therefore, further studies will be conducted to explore Sewing from 2-side to n-side of a 3D target.

Second is that, for the empirical study, we only set vertical and horizontal (two commonly mentioned conditions in previous research). Whereas, it should be noted that 3D objects may be placed with any orientation. Therefore, in future work, Sewing performance will be evaluated with various tilt degrees. Moreover, there may be scenarios where targets are distributed with different level of density in 3D virtual environment (similar to conditions in [49]), which may bring challenges during Sewing. Therefore, as part of the future work, studies regarding the target density will also be conducted to explore its influence on the Sewing performance.

Third is that, we did not consider the penetration angle (between the spatial penetration path and its projection part on the penetrated surface) of the virtual mallet tip in our previous studies. This angle can be used to provide additional information or trigger other functionality when penetration happens. Therefore, we will also explore potential input channels in our future work.

Additionally, for more advanced VR GUI designs, interaction scenarios would be more sophisticated: 1) targets may be placed in different 3D planes and 2) penetration side and direction may be difficult to identify when Sewing happens with occlusions among 3D targets. Therefore, series of studies will be conducted to explore different visualization methods of the penetration process and various visual feedback approaches to clarify and guide users to finish the Sewing on the intended target and surface.

8. Conclusion

To improve target selection coherence among multiple targets in VR, we present an empirical evaluation of Sewing, a successive target selection method in VR applications. By leveraging the penetrable feature of intangible displays and the real-life sewing metaphor, users can select multiple targets successively and seamlessly through penetrating in from one target and out directly towards another. Our evaluation results showed that, when sequentially selecting multiple targets, Sewing regarded all targets as nodes of a single penetration movement and achieved coherent and rapid selections. We further implemented an instance of Sewing (named SewTyping) and validated its ability to facilitate users' VR text entry performance. Our findings not only provide a method to improve the coherence of VR target selection towards multiple targets but also an unconventional perspective to explore the characteristics of intangible interfaces for better interaction performance.

9. Acknowledgment

We thank all members of the Center for Human-Engaged Computing (CHEC) at Kochi University of Technology for their support. We also thank Zengyi Han for his contributions to the initial stage of discussions in this work.

References

[1] 360marketupdates. *2019-2024 Global and Regional Augmented Reality and Virtual Reality Industry Production, Sales and Consumption Status and Prospects Professional Market Research Report*. 2019. URL: `https://www.360marketupdates.com/2019-2024-global-and-regional-augmented-reality-and-virtual-reality-industry-production-sales-and-consumption-status-and-prospects-professional-market-research-report-13733991`.

[2] Johnny Accot and Shumin Zhai. "More than dotting the i's- Foundations for crossing-based interfaces". In: *Conference on Human Factors in Computing Systems - Proceedings*. 2002. ISBN: 1581134533. DOI: `10.1145/503376.503390`.

[3] Georg Apitz and François Guimbretière. "CrossY: A Crossing-Based Drawing Application". In: *Proceedings of the 17th Annual ACM Symposium on User Interface Software and Technology*. UIST '04. Santa Fe, NM, USA: Association for Computing Machinery, 2004, pp. 3–12. DOI: `10.1145/1029632.1029635`. URL: `https://doi.org/10.1145/1029632.1029635`.

[4] Ferran Argelaguet and Carlos Andujar. "A survey of 3D object selection techniques for virtual environments". In: *Computers Graphics* 37.3 (2013), pp. 121–136. ISSN: 0097-8493. DOI: https://doi.org/10.1016/j.cag.2012.12.003. URL: http://www.sciencedirect.com/science/article/pii/S0097849312001793.

[5] Marc Baloup, Thomas Pietrzak, and Géry Casiez. "RayCursor: A 3D Pointing Facilitation Technique Based on Raycasting". In: *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. CHI '19. Glasgow, Scotland Uk: Association for Computing Machinery, 2019, pp. 1–12. ISBN: 9781450359702. DOI: 10.1145/3290605.3300331. URL: https://doi.org/10.1145/3290605.3300331.

[6] Angela Benavides, Rajiv Khadka, and Amy Banic. "Physically-Based Bimanual Volumetric Selection for Immersive Visualizations". In: *Virtual, Augmented and Mixed Reality. Multimodal Interaction*. Ed. by Jessie Y.C. Chen and Gino Fragomeni. Cham: Springer International Publishing, 2019, pp. 183–195. ISBN: 978-3-030-21607-8.

[7] Costas Boletsis and Stian Kongsvik. "Controller-based Text-input Techniques for Virtual Reality: An Empirical Comparison". In: *International Journal of Virtual Reality* (2019). ISSN: 1081-1451. DOI: 10.20870/ijvr.2019.19.3.2917.

[8] Doug Bowman et al. "Using pinch gloves (tm) for both natural and abstract interaction techniques in virtual environments". In: (2001).

[9] John Brooke. "SUS: a retrospective". In: *Journal of Usability Studies* (2013). ISSN: 1931-3357.

[10] Li-Wei Chan et al. "Touching the Void: Direct-Touch Interaction for Intangible Displays". In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '10. Atlanta, Georgia, USA: Association for Computing Machinery, 2010, pp. 2625–2634. DOI: 10.1145/1753326.1753725. URL: https://doi.org/10.1145/1753326.1753725.

[11] Inrak Choi et al. "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*. CHI '18. Montreal QC, Canada: Association for Computing Machinery, 2018, pp. 1–13. ISBN: 9781450356206. DOI: 10.1145/

3173574.3174228. URL: https://doi.org/10.1145/3173574.3174228.

[12] Nathan Cournia, John D. Smith, and Andrew T. Duchowski. "Gaze- vs. Hand-Based Pointing in Virtual Environments". In: *CHI '03 Extended Abstracts on Human Factors in Computing Systems*. CHI EA '03. Ft. Lauderdale, Florida, USA: Association for Computing Machinery, 2003, pp. 772–773. ISBN: 1581136374. DOI: 10.1145/765891.765982. URL: https://doi.org/10.1145/765891.765982.

[13] Cutiekeys. *NormalVR/CutieKeys*. 2017. URL: https://github.com/NormalVR/CutieKeys/.

[14] Seyed Amir Ahmad Didehkhorshid and Robert J Teather. "Selection Performance Using a Scaled Virtual Stylus Cursor in VR". In: (2020).

[15] Pierre Dragicevic. "Combining Crossing-Based and Paper-Based Interaction Paradigms for Dragging and Dropping between Overlapping Windows". In: *Proceedings of the 17th Annual ACM Symposium on User Interface Software and Technology*. UIST '04. Santa Fe, NM, USA: Association for Computing Machinery, 2004, pp. 193–196. ISBN: 1581139578. DOI: 10.1145/1029632.1029667. URL: https://doi.org/10.1145/1029632.1029667.

[16] Anna Maria Feit, Daryl Weir, and Antti Oulasvirta. "How we type: Movement strategies and performance in everyday typing". In: *Conference on Human Factors in Computing Systems - Proceedings*. 2016. ISBN: 9781450333627. DOI: 10.1145/2858036.2858233.

[17] P. M. Fitts. "The information capacity of the human motor system in controlling the amplitude of movement". In: *Journal of Experimental PSychology* 74 (1954), pp. 381–391.

[18] Deanna H. Gates and Jonathan B. Dingwell. "The effects of neuromuscular fatigue on task performance during repetitive goal-directed movements". In: *Experimental Brain Research* (2008). ISSN: 00144819. DOI: 10.1007/s00221-008-1326-8.

[19] Tovi Grossman and Ravin Balakrishnan. "The Design and Evaluation of Selection Techniques for 3D Volumetric Displays". In: *Proceedings of the 19th Annual ACM Symposium on User Interface Software and Technology*. UIST '06. Montreux, Switzerland: Association for Computing Machinery, 2006, pp. 3–12. ISBN: 1595933131. DOI: 10.1145/

1166253.1166257. URL: https://doi.org/10.1145/1166253.1166257.

[20] Jens Grubert et al. “The Office of the Future: Virtual, Portable, and Global”. In: *IEEE Computer Graphics and Applications* (2018). ISSN: 15581756. DOI: 10.1109/MCG.2018.2875609.

[21] Jie Guo et al. “Evaluation of maslows hierarchy of needs on long-term use of HMDs - A case study of office environment”. In: *26th IEEE Conference on Virtual Reality and 3D User Interfaces, VR 2019 - Proceedings*. 2019. ISBN: 9781728113777. DOI: 10.1109/VR.2019.8797972.

[22] International Organization For Standardization. *ISO 9241-9*. 2000.

[23] Akira Ishii et al. “FistPointer: Target Selection Technique Using Mid-Air Interaction for Mobile VR Environment”. In: *Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems*. CHI EA ’17. Denver, Colorado, USA: Association for Computing Machinery, 2017, p. 474. ISBN: 9781450346566. DOI: 10.1145/3027063.3049795. URL: https://doi.org/10.1145/3027063.3049795.

[24] Jason Jerald. *The VR Book: Human-Centered Design for Virtual Reality*. Association for Computing Machinery and Morgan Claypool, 2015. ISBN: 9781970001129.

[25] Joseph J. LaViola Jr. *3D User Interfaces: Theory and Practice (2nd Edition) (Usability)*. Addison-Wesley Professional, Apr. 2017. ISBN: 0134034325. URL: https://www.xarg.org/ref/a/0134034325/.

[26] Per Ola Kristensson and Keith Vertanen. “Performance comparisons of phrase sets and presentation styles for text entry evaluations”. In: *International Conference on Intelligent User Interfaces, Proceedings IUI*. 2012. ISBN: 9781450310482. DOI: 10.1145/2166966.2166972.

[27] John Lucas et al. “Design and Evaluation of 3D Multiple Object Selection Techniques”. In: (Jan. 2005).

[28] Yuexing Luo and Daniel Vogel. “Crossing-Based Selection with Direct Touch Input”. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI ’14. Toronto, Ontario, Canada: Association for Computing Machinery, 2014, pp. 2627–2636. DOI: 10.1145/2556288.2557397. URL: https://doi.org/10.1145/2556288.2557397.

[29] Yuexing Luo and Daniel Vogel. "Pin-and-Cross: A Unimanual Multitouch Technique Combining Static Touches with Crossing Selection". In: *Proceedings of the 28th Annual ACM Symposium on User Interface Software Technology*. UIST '15. Charlotte, NC, USA: Association for Computing Machinery, 2015, pp. 323–332. ISBN: 9781450337793. DOI: 10.1145/2807442.2807444. URL: https://doi.org/10.1145/2807442.2807444.

[30] Cuixia Ma et al. "An adaptive sketching user interface for education system in virtual reality". In: *ITME2009 - Proceedings 2009 IEEE International Symposium on IT in Medicine and Education*. 2009. ISBN: 9781424439294. DOI: 10.1109/ITIME.2009.5236314.

[31] I. Scott MacKenzie. "Fitts' Law as a Research and Design Tool in Human-Computer Interaction". In: *Hum.-Comput. Interact.* 7.1 (Mar. 1992), pp. 91–139. ISSN: 0737-0024. DOI: 10.1207/s15327051hci0701_3. URL: https://doi.org/10.1207/s15327051hci0701_3.

[32] David E Meyer et al. "Optimality in human motor performance: ideal control of rapid aimed movements." In: *Psychological review* 95.3 (1988), p. 340.

[33] Mark R. Mine, Frederick P. Brooks, and Carlo H. Sequin. "Moving Objects in Space: Exploiting Proprioception in Virtual-Environment Interaction". In: *Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques*. SIGGRAPH '97. USA: ACM Press, 1997, pp. 19–26. ISBN: 0897918967. DOI: 10.1145/258734.258747. URL: https://doi.org/10.1145/258734.258747.

[34] R. A. Montano-Murillo et al. "Slicing-Volume: Hybrid 3D/2D Multi-target Selection Technique for Dense Virtual Environments". In: *2020 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*. 2020, pp. 53–62.

[35] Cuong Nguyen et al. "CollaVR: Collaborative in-headset review for VR video". In: *UIST 2017 - Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology*. 2017. ISBN: 9781450349819. DOI: 10.1145/3126594.3126659.

[36] Charles Perin and Pierre Dragicevic. "Manipulating Multiple Sliders by Crossing". In: *Proceedings of the 26th Conference on l'Interaction Homme-Machine*. IHM '14. Villeneuve d'Ascq, France: Association for

Computing Machinery, 2014, pp. 48–54. ISBN: 9781450329354. DOI: 10.1145/2670444.2670449. URL: https://doi.org/10.1145/2670444.2670449.

[37] Ivan Poupyrev et al. "The Go-Go Interaction Technique: Non-Linear Mapping for Direct Manipulation in VR". In: *Proceedings of the 9th Annual ACM Symposium on User Interface Software and Technology*. UIST '96. Seattle, Washington, USA: Association for Computing Machinery, 1996, pp. 79–80. ISBN: 0897917987. DOI: 10.1145/237091.237102. URL: https://doi.org/10.1145/237091.237102.

[38] Adrian Ramcharitar and Robert J. Teather. "EZCursorVR: 2D Selection with Virtual Reality Head-Mounted Displays". In: *Proceedings of the 44th Graphics Interface Conference*. GI '18. Toronto, Canada: Canadian Human-Computer Communications Society, 2018, pp. 123–130. ISBN: 9780994786838. DOI: 10.20380/GI2018.17. URL: https://doi.org/10.20380/GI2018.17.

[39] R. William Soukoreff and I. Scott MacKenzie. "Metrics for text entry research: An evaluation of MSD and KSPC, and a new unified error metric". In: *Conference on Human Factors in Computing Systems - Proceedings*. 2003.

[40] R. William Soukoreff and I. Scott MacKenzie. "Towards a Standard for Pointing Device Evaluation, Perspectives on 27 Years of Fitts' Law Research in HCI". In: *Int. J. Hum.-Comput. Stud.* 61.6 (Dec. 2004), pp. 751–789. ISSN: 1071-5819. DOI: 10.1016/j.ijhcs.2004.09.001. URL: https://doi.org/10.1016/j.ijhcs.2004.09.001.

[41] Marco Speicher et al. "Selection-based text entry In Virtual Reality". In: *Conference on Human Factors in Computing Systems - Proceedings*. 2018. ISBN: 9781450356206. DOI: 10.1145/3173574.3174221.

[42] Frank Steinicke, Timo Ropinski, and Klaus Hinrichs. "OBJECT SELECTION IN VIRTUAL ENVIRONMENTS USING AN IMPROVED VIRTUAL POINTER METAPHOR". In: *Computer Vision and Graphics: International Conference, ICCVG 2004, Warsaw, Poland, September 2004, Proceedings*. Ed. by K. Wojciechowski et al. Dordrecht: Springer Netherlands, 2006, pp. 320–326. ISBN: 978-1-4020-4179-2. DOI: 10.1007/1-4020-4179-9_46. URL: https://doi.org/10.1007/1-4020-4179-9_46.

[43] Rasmus Stenholt. "Efficient Selection of Multiple Objects on a Large Scale". In: *Proceedings of the 18th ACM Symposium on Virtual Reality Software and Technology*. VRST '12. Toronto, Ontario, Canada: Association for Computing Machinery, 2012, pp. 105–112. ISBN: 9781450314695. DOI: 10.1145/2407336.2407357. URL: https://doi.org/10.1145/2407336.2407357.

[44] Ahmed N. Sulaiman and Patrick Olivier. "Attribute Gates". In: *Proceedings of the 21st Annual ACM Symposium on User Interface Software and Technology*. UIST '08. Monterey, CA, USA: Association for Computing Machinery, 2008, pp. 57–66. ISBN: 9781595939753. DOI: 10.1145/1449715.1449726. URL: https://doi.org/10.1145/1449715.1449726.

[45] Robert J. Teather and Wolfgang Stuerzlinger. "Factors Affecting Mouse-Based 3D Selection in Desktop VR Systems". In: *Proceedings of the 3rd ACM Symposium on Spatial User Interaction*. SUI '15. Los Angeles, California, USA: Association for Computing Machinery, 2015, pp. 10–19. ISBN: 9781450337038. DOI: 10.1145/2788940.2788946. URL: https://doi.org/10.1145/2788940.2788946.

[46] Robert J. Teather, Wolfgang Stuerzlinger, and Andriy Pavlovych. "Fishtank Fitts: A Desktop VR Testbed for Evaluating 3D Pointing Techniques". In: *CHI '14 Extended Abstracts on Human Factors in Computing Systems*. CHI EA '14. Toronto, Ontario, Canada: Association for Computing Machinery, 2014, pp. 519–522. ISBN: 9781450324748. DOI: 10.1145/2559206.2574810. URL: https://doi.org/10.1145/2559206.2574810.

[47] C. A. Terzuolo and P. Viviani. "Determinants and characteristics of motor patterns used for typing". In: *Neuroscience* (1980). ISSN: 03064522. DOI: 10.1016/0306-4522(80)90188-8.

[48] Huawei Tu et al. "Crossing-based selection with virtual reality head-mounted displays". In: *Conference on Human Factors in Computing Systems - Proceedings*. 2019. ISBN: 9781450359702. DOI: 10.1145/3290605.3300848.

[49] Lode Vanacken, Tovi Grossman, and Karin Coninx. "Exploring the Effects of Environment Density and Target Visibility on Object Selection in 3D Virtual Environments". In: *2007 IEEE Symposium on 3D User Interfaces*. 2007. DOI: 10.1109/3DUI.2007.340783.

[50] Keith Vertanen and Per Ola Kristensson. “A Versatile Dataset for Text Entry Evaluations Based on Genuine Mobile Emails”. In: *Proceedings of the 13th International Conference on Human Computer Interaction with Mobile Devices and Services*. MobileHCI ’11. New York, NY, USA: ACM, 2011, pp. 295–298. ISBN: 978-1-4503-0541-9. DOI: 10.1145/2037373.2037418. URL: http://doi.acm.org/10.1145/2037373.2037418.

[51] Colin Ware. *Information Visualization: Perception for Design (Interactive Technologies)*. Morgan Kaufmann, June 2012. ISBN: 0123814642. URL: https://www.xarg.org/ref/a/0123814642/.

[52] Jacob O. Wobbrock and Krzysztof Z. Gajos. “Goal Crossing with Mice and Trackballs for People with Motor Impairments: Performance, Submovements, and Design Directions”. In: *ACM Trans. Access. Comput.* 1.1 (May 2008). ISSN: 1936-7228. DOI: 10.1145/1361203.1361207. URL: https://doi.org/10.1145/1361203.1361207.

[53] Yukang YAN et al. “Gesture-based target acquisition in virtual and augmented reality”. In: *Virtual Reality Intelligent Hardware* 1.3 (2019), pp. 276–289. ISSN: 2096-5796. DOI: https://doi.org/10.3724/SP.J.2096-5796.2019.0007. URL: http://www.sciencedirect.com/science/article/pii/S2096579619300233.

[54] Yukang Yan et al. “Eyes-Free Target Acquisition in Interaction Space around the Body for Virtual Reality”. In: *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. CHI ’18. Montreal QC, Canada: Association for Computing Machinery, 2018, pp. 1–13. ISBN: 9781450356206. DOI: 10.1145/3173574.3173616. URL: https://doi.org/10.1145/3173574.3173616.

[55] Yukang Yan et al. “HeadCross: Exploring Head-Based Crossing Selection on Head-Mounted Displays”. In: *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 4.1 (Mar. 2020). DOI: 10.1145/3380983. URL: https://doi.org/10.1145/3380983.

[56] Xin Yi et al. “ATK: Enabling ten-finger freehand typing in air based on3D hand tracking data”. In: *UIST 2015 - Proceedings of the 28th Annual ACM Symposium on User Interface Software and Technology*. 2015. ISBN: 9781450337793. DOI: 10.1145/2807442.2807504.

[57] Chun Yu et al. "Tap, dwell or gesture?: Exploring head-based text entry techniques for HMDs". In: *Conference on Human Factors in Computing Systems - Proceedings*. 2017. ISBN: 9781450346559. DOI: 10.1145/3025453.3025964.

[58] Difeng Yu et al. "Modeling Endpoint Distribution of Pointing Selection Tasks in Virtual Reality Environments". In: *ACM Trans. Graph.* 38.6 (Nov. 2019). ISSN: 0730-0301. DOI: 10.1145/3355089.3356544. URL: https://doi.org/10.1145/3355089.3356544.

[59] Shumin Zhai, William Buxton, and Paul Milgram. "The "Silk Cursor": Investigating Transparency for 3D Target Acquisition". In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '94. Boston, Massachusetts, USA: Association for Computing Machinery, 1994, pp. 459–464. ISBN: 0897916506. DOI: 10.1145/191666.191822. URL: https://doi.org/10.1145/191666.191822.