PinchCatcher: Enabling Multi-selection for Gaze+Pinch

Jinwook Kim
jinwook.kim31@kaist.ac.kr
Graduate School of Culture Technology
KAIST
Daejeon, South Korea

Sangmin Park
psmdc0714@kaist.ac.kr
Graduate School of Culture Technology
KAIST
Daejeon, South Korea

Qiushi Zhou
qiushi.zhou@cs.au.dk
Aarhus University
Aarhus, Denmark

Mar Gonzalez-Franco
margonzalezfranco@gmail.com
Google
Seattle, WA, USA

Jeongmi Lee*
jeongmi@kaist.ac.kr
Graduate School of Culture Technology
KAIST
Daejeon, South Korea

Ken Pfeuffer*
ken@cs.au.dk
Aarhus University
Aarhus, Denmark

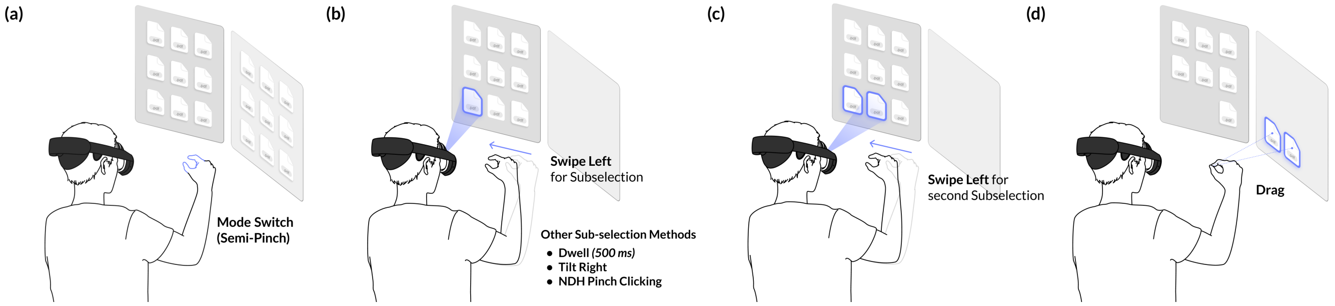

Figure 1: We simplify multi-selection for eye and hand interfaces in extended reality (XR) by utilizing the pre-pinch phase (a) —when users initiate but do not complete a pinch gesture— to contextualize upcoming actions. In this quasi-mode, users can gaze at multiple objects, subselecting them for the subsequent full-pinch command (b-c), allowing for intuitive, pre-pinch multi-selection before initiating gesture operations like drag and drop (d).

ABSTRACT

This paper investigates multi-selection in XR interfaces based on eye and hand interaction. We propose enabling multi-selection using different variations of techniques that combine gaze with a semi-pinch gesture, allowing users to select multiple objects, while on the way to a full-pinch. While our exploration is based on the semi-pinch mode for activating a quasi-mode, we explore four methods for confirming subselections in multi-selection mode, varying in effort and complexity: dwell-time (SemiDwell), swipe (SemiSwipe), tilt (SemiTilt), and non-dominant hand input (SemiNDH), and compare them to a baseline technique. In the user study, we evaluate their effectiveness in reducing task completion time, errors, and effort. The results indicate the strengths and weaknesses of each technique, with SemiSwipe and SemiDwell as the most preferred methods by participants. We also demonstrate their utility in file managing and RTS gaming application scenarios. This study provides valuable insights to advance 3D input systems in XR.

CCS CONCEPTS

• **Human-centered computing → Interaction techniques; Mixed / augmented reality; Virtual reality; User studies; Interaction design theory, concepts and paradigms.**

KEYWORDS

Extended Reality, Selection, Grouping, Gaze, Gestures, Eye-Hand interaction

ACM Reference Format:
Jinwook Kim, Sangmin Park, Qiushi Zhou, Mar Gonzalez-Franco, Jeongmi Lee, and Ken Pfeuffer. 2025. PinchCatcher: Enabling Multi-selection for Gaze+Pinch. In *CHI Conference on Human Factors in Computing Systems (CHI '25), April 26-May 1, 2025, Yokohama, Japan.* ACM, New York, NY, USA, 16 pages. https://doi.org/10.1145/3706598.3713530

1 INTRODUCTION

The gaze + pinch has the potential to become as widely available in virtual reality (VR) as touch input in tablets and smartphones, as a direct natural interaction [46, 47]. Still, it has several shortcomings, like the ability for preshaping of our hand [38], competing

*Co-corresponding author

CHI '25, April 26-May 1, 2025, Yokohama, Japan

ACM ISBN 979-8-4007-1394-1/25/04
https://doi.org/10.1145/3706598.3713530

attentional demands of gaze in augmented reality (AR) [43], and a limited throughput in multi-selection scenarios. On smartphones, we have moved away from having to tap on every item or photo in our gallery to select fast, by adding hold-and-swipe type of gestures. We need our equivalent for gaze + pinch.

Multi-selection techniques render the process of selecting multiple items such as files or images more efficient [37, 77, 78]. Multi-selection can be enabled through a *persistent* mode, such as long-pressing on a touchscreen, that stays active until manually turned off [53]. Alternatively, a *quasi-mode* (or spring loaded [16]), such as when holding the CTRL key in a mouse and keyboard-based UI, which is suitable for spontaneous multi-selections.

Although multi-selection has matured for traditional computing devices, it is open to how eye- and hand-tracked UIs in extended reality (XR) can support such a mechanism. Many XR UIs are based on the canonical pinch gesture, offering dexterous and natural ways to manipulate single objects. Multi-selection is supported via persistent modes (e.g., special gestures like long-pinch or dedicated UI buttons [11, 28, 37, 55]), but quasi-mode-based multi-selection features are difficult to integrate. To date, only Shi et al. have explored this area in the context of gesture-only UIs [56]. Their approach introduces additional gestures that indeed enable mode-switching but it trades it with higher effort as users need additional cognitive resources to learn and remember them [40, 56]. We propose a different approach by designing for multi-selection without departing from the intuitive pinch-based input paradigm.

People naturally look at objects of interest in point-select tasks environments [36, 49, 69, 80]. Modern headsets (e.g., Hololens 2, Quest Pro, Vision Pro) exploit this behavior, offering a multimodal UI based on user inputs interpreted through gaze and pinch gestures [8, 47, 72, 73]–we posit that this behavior will hold specifically in the context of manipulating multiple objects at once. Before the manipulation commences, users are likely to look at the objects they want to manipulate, giving away which and how many targets are to be manipulated. As well, users are likely to prepare their pinch gesture in anticipation of the subsequent manipulation [83]. Taken together, we experiment with this dual phenomenon for implicitly specifying all objects of interest with gaze just before one engages in a pinch gesture.

We propose PinchCatcher, a new approach for multi-selection that allows subselecting multiple targets for single pinch gestures in eye- and hand-tracked XR. The idea is to integrate gaze pointing with a semi-pinch gesture [83] that acts as a quasi-mode. This state occurs when users maintain their fingers in a partially pinched position, hovering between fully extended and fully flexed. Figure 1 illustrates the concept. When users maintain a semi-pinch posture (a), the system interprets this signal as a mode-switch to a pre-pinch state, which allows them to contextualize the subsequent full gesture. In this state, users can select multiple objects by simply gazing at each object of interest (b-c). When complete, a pinch-contact gesture manipulates all objects at once (d).

This leads us to our main research question: how can users, while holding a semi-pinch state, add more targets to the selection by gaze without accidentally including unwanted items (i.e., Midas Touch problem [23, 24, 42])? As users are only briefly holding a semi-pinch gesture to contextualize their subsequent action, it needs to balance simplicity of input with low cognitive and physical effort, all while being sufficiently robust to clarify intent. We explore four distinct methods for triggering subselection while users maintain the semi-pinch state, using gaze as the primary pointer: (1) SemiDwell, a low-effort technique based on dwell time with the eyes alone; (2) SemiSwipe and (3) SemiTilt, both utilizing gestures from the same hand; and (4) SemiNDH, where the non-dominant hand (NDH) performs pinch gestures while the dominant hand (DH) maintains the semi-pinch.

We present a user study that compares the four techniques against a baseline (FullDH) that closely resembles the CTRL key approach of desktop UI [27, 74]. Here, the user holds a pinch gesture of the NDH, and normal pinch gestures of the DH will add to the multi-selection. These multi-selection techniques are tested in a serial selection task [3, 77] where users select each object individually. This task examines fundamental subselection capabilities, which are critical for guiding future developments, such as area selection [30, 57, 81]. Objects were arranged in a 2D grid in space, reflecting common layouts in current XR UIs. Participants were tasked with selecting 2, 4, or 6 targets, focusing on our quasi-mode techniques, which are intended for scenarios involving a few targets, rather than a large number where persistent modes might be more effective. Based on the findings, we demonstrate the idea of using PinchCatcher in two applications: file management, which is a fundamental task when using the computer, and real-time strategy (RTS) game for usage in the 3D environment.

The results of the study show the following findings:

- There was no significant difference in task completion time across the techniques.
- Participants selected more distractors with the FullDH technique than all other techniques but could correct them before finalizing the selection, leading to the lowest error rate.
- The SemiSwipe technique resulted in fewer distractor selections than SemiDwell and SemiNDH, and overall fewer errors than SemiDwell and SemiTilt.
- SemiNDH led to more perceived physical effort than SemiDwell and FullDH.
- Most participants preferred SemiDwell (9) and SemiSwipe (8).

In summary, our contributions are as follows: First, we introduce PinchCatcher, an interaction concept for multi-selection in XR that combines the benefits of a semi-pinch quasi-mode with rapid eye movement-based input. Second, we propose four distinct methods for confirming subselections during the semi-pinch state, each with its strengths and limitations. Third, we present a user study that provides insights into user performance and experience, demonstrating the effectiveness of our techniques compared to a bimanual approach. Fourth, we offer application probes, such as photo management and RTS gaming, to showcase use cases and provide guidance for implementing PinchCatcher. Our work provides valuable insights for future 3D input solutions and highlights o new opportunities to advance eye- and hand-tracked interaction in XR.

2 RELATED WORK

2.1 Multi-selection Interaction in Representative Platforms

Multi-selection especially provides convenience for performing the same task on multiple targets (e.g., dragging, copying, and deleting) [37]. The multi-selection consists of two steps: mode switching and triggering subselection. Users first switch to multi-selection mode from single selection mode. Then, they target and confirm the grouping as they did in a single selection. Additionally, there are serial and parallel ways of grouping [37, 59]. Serial grouping is a method that selects target objects individually, while parallel methods group multiple targets simultaneously. For the serial way, users individually select the targets with a basic selection input method such as a mouse click, direct touch, and ray interaction for grouping [74, 77]. In terms of the parallel way, users utilized the predefined 3D cones or spotlights shape region [37, 56, 64] or generate a region by selecting two points with a lasso or a selection box tool for grouping [57, 77, 81].

In the desktop environment (Figure 2), users perform multi-selection by combining keyboard and mouse inputs [27, 74]. For instance, users press the CTRL or SHIFT key on the keyboard to activate multi-selection as a quasi-mode and group targets with the mouse or arrow button input. Regarding multi-selection in the touch screen (Figure 2), users touch one of the components and dwell on activating the persistent multi-selection mode or enabling it with the mode switching button UI [5, 10, 44, 52].

Overall, two types of multi-selection activation methods have been utilized in the previous platforms: a quasi-mode that maintains the input for the mode switch while grouping, and a persistent mode that separates the mode switch and grouping process. Users may lack awareness of the selection status in the persistent mode state, leading to unintended selection with different modes. To illustrate, users could attempt to perform a multi-selection but were inadvertently in a single selection mode. Consequently, they would have to change the mode and perform the multi-selection from the beginning. In contrast, since users must maintain their actions in quasi-mode-based techniques, it could prevent potential errors by supporting their awareness of the selection status [16].

Similarly, a VR environment with a physical controller uses both types of methods by using physical button/joystick inputs or menu UI to change the selection mode [11, 77, 82]. In terms of direct physical input, Wu et al. explored the performance and usability of the various multi-selection methods, directly touching each object with the controller serially or making an arbitrary area and modifying the size with the controller movement for parallel selection [77]. The menu UI for enabling multi-selection mode is utilized in the recent VR and AR applications (i.e., ShapesXR, Google Blocks) [28, 55]. For instance, in the ShapeXR application, the user has to press the button on the controller to activate the pop-up menu and select the multi-selection option with the ray-casting from the controller. Then they select which multi-selection technique to use (i.e., serial group selection, lasso area selection) and begin grouping the targets.

Several studies have proposed techniques that utilized additional interfaces (i.e., virtual or physical tablets) with the physical controller to assist multi-selection in 3D environments [37, 39]. For instance, Lucas et al. proposed a method that uses a virtual tablet that shows a camera view image of a specific point and can draw an area on the image for multi-selection [37]. There are also techniques based on the metaphor of actions from daily life [21, 34]. Li et al. proposed a multi-selection technique based on sewing action with the controller [34]. The user evaluation results showed that these techniques had improved the performance of multi-selection in specific contexts (i.e., typing, 3D modeling). However, it requires multiple button inputs to change the selection mode and enable the multi-selection technique. This process could slow down the performance of grouping targets and switching between modes, which makes it hard for this technique to be used in broader XR contexts, such as game and remote co-working applications.

2.2 Multi-selection for Hand and Eye Interaction in XR

Gaze input enables fast, natural target acquisition [7, 25], while hand gestures are highly suitable for expressive commands [4, 22, 75]. Their multimodal fusion—where the eyes target and the hands 'click'—has shown great potential across various human-computer interaction paradigms, including mouse-operated systems [24, 80], touch devices [44, 66], pen computing [45], and 3D gesture-controlled desktops [7, 72]. Modern XR Head Mounted Displays (HMDs) increasingly embed eye tracking sensors which have numerous applications [50], such as for tasks like one-handed menu control [48], 3D sketching [70], and hands-free accessibility [60]. A key interaction model, gaze + pinch, follows carefully crafted design principles that could be utilized in universal XR operating systems [46, 47], as exemplified by the Vision Pro HMD.

Accordingly, we build upon this interaction model by experimenting with multi-selection techniques. Studies have proposed multi-selection techniques for the hand-based interaction contexts [30, 33, 56, 81], by using both hand pinch gesture [30] or ray from multiple finger [81] to make an arbitrary area for group selection. Complementary to the work on multi-selection, Surale et al. explored mode-switching through various gestures such as a fist, palm, pinching, and pointing in both DH and NDH [67]. They concluded that the pinch with DH is a viable option for a controller button for mode-switching, which inspired us to take a deeper look at this gesture. Hu et al. investigated using gaze input for mode-switching on a pen and tablet device [20]. They found that gaze alone can provide an efficient mode-switch with low effort, however, a multimodal variant where the user fixated on a distinct UI element coupled with a manual trigger led to more robust mode-switches [20]. Inspired by this, our interaction design considers how gaze and pinch gestures can be used together for a mode switch to a multi-selection mode.

Multi-selection techniques that combine eye gaze with hand gestures are rarely considered in the literature. For instance, a closely related work is Shi et al. work on region selection where the user defines two points that form the diagonal of a region [57], which in theory can be used for area selection. They evaluated four techniques: Gaze&Finger, Gaze&Pinch, Pinch, and Eyeblink, and found that both unimodal approaches work best. Our work is complementary, as we consider a new way of semi-pinch as a mode-switch, to mode a pinch gesture. This affords flexible gestures without altering the default operation of pinch-click, drag, and so on, and avoids

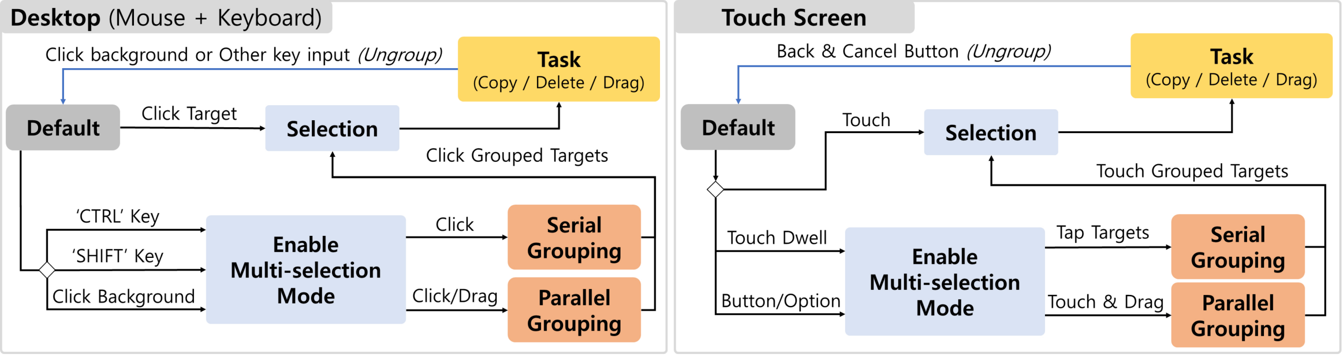

Figure 2: A diagram of a multi-selection process in 2D desktop and touch screen environments.

overlap with the gestures employed for other functions [2, 46]. Further, we do not consider eyes-only approaches as a mode-switch, as the active use of the eyes for UI manipulations alters the natural division of labor between the eye and hand modalities [46]. As well, if the interaction relies too much on the gaze, it could cause eye fatigue and errors [18]. When designing interaction with gaze and hand, it is essential to design for the natural roles of each modality and exploit their synergies [29, 80]– which is here, that users naturally focus on the candidates for multi-selection, coupled with an explicit pre-pinch event.

3 DESIGN OF SERIAL MULTI-SELECTION TECHNIQUES FOR GAZE + PINCH INTERACTION

We describe the design of multi-selection techniques for eye-hand interaction systems that take natural gaze behavior into account and that are compatible with the intuitive pinch-based manipulation paradigm. Here we first lay out our design rationale for choosing serial multi-selection and one-handed interaction. We then lay out our four multi-selection techniques. We also describe a mouse and keyboard-inspired pinch technique (FullDH) as a baseline for comparison to the semi-pinch-based techniques.

3.1 Design Considerations

3.1.1 Serial Way of Multi-selection. Among the two ways of multi-selection (i.e., serial and parallel), the parallel way seems to be faster since it is capable of grouping multiple targets with one action. However, previous studies have proposed that the performance and usability of serial and parallel ways significantly differ depending on the context [37, 77]. For instance, if a large number of targets are distributed close to each other, the parallel methods have an advantage over the serial methods. On the other hand, if the number of targets is small (below 10) and targets are situated near distractors, the serial methods demonstrate enhanced performance. Both methods are important for an efficient and comfortable multi-selection experience, but parallel methods require structured environments that are less general in the XR context. Thus, we focused the scope of the current study on the serial grouping methods.

3.1.2 Maintain One-hand Interaction for Gaze + Pinch. In terms of designing a multi-selection method that could maximize the advantage of gaze, it is essential to consider the connectivity with the functions (e.g., selection, dragging, and resizing) that were performed with the Gaze + Pinch previously (cf. '*Flexible Gesture*' [46]). For instance, a single pinch gesture with the DH is intuitively assigned to a single object selection, and a pinch with both hands for resizing or rotating a selected object. Adding a new hand gesture instead of a pinch might require users to learn additional gestures or actions. In addition, using the other hand limits the freedom of the hand and the flexibility of other functions after or during multi-selection. Consider this example: if multiple selections are possible with a single hand, users can perform other tasks in parallel or tasks that could support the right-hand multiple selections (e.g., move objects that occlude target, scroll the list to search for more targets). Thus, we deliberately constrain our design to a one-handed interaction, retaining compatibility with the default interaction model of pinch-based manipulations.

3.2 Semi-Pinch State for Mode Switching Method

Based on the design considerations, we propose a mode-switching method using a semi-pinch gesture. The semi-pinch gesture was used in a previous study as a preselection state for displaying lens interface and adopting a control-display gain on hand to enable precise selection of small objects [83]. Since the *PinchLens* technique uses hand gestures for both selection and confirmation, it requires users to keep attention on tracking the pinch status (i.e., the distance between index and thumb fingertip) to manipulate the lens interface while conducting the task, and it is susceptible to the Heisenberg issues [76]. In contrast, the Gaze + Pinch interaction separates these roles to "the eyes select, the hands manipulate" [46], where the attention is not on the hand gesture and it is more robust to the Heisenberg issues [76]. When using semi-pinch and gaze, it must be clear to the user how to enter and exit the semi-pinch status without looking at the physical hand, as the eyes' attentional resources are reserved for the targets before finalizing a selection.

As illustrated in Figure 3, we mainly utilized three states of a pinch: full pinch, semi pinch, and full release pinch. When the full pinch is triggered, the grouping state ends, and users can start manipulating the grouped objects. In the semi-pinch state, the mode was changed to a multi-selection mode, and the full-release pinch

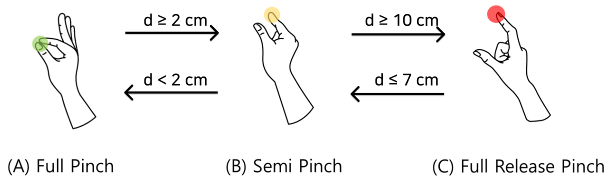

Figure 3: Illustration of the three hand-pinch states used to design the multi-selection method. Each state was detected based on the distance between the index and thumb fingertips. The indicator for each state is shown on the index fingertip. (A) Full pinch is detected when both fingertips are touching, and the indicator is highlighted in green. Users were able to interact with the selected objects in this state. (B) Semi-pinch is used to enable multi-selection mode and is activated when the fingertip distance is between 2 and 7 cm, as shown in the yellow indicator. (C) Full-release pinch is a state when the distance is over 10 cm and is indicated in red. It disables the grouping of all objects.

made all grouped objects to an ungrouped state. We implemented the hand gesture detection using Meta Hand Pose Detection SDK and added the detection threshold between the full-release pinch and semi-pinch status to prevent the false activation of the ungrouping function. The distance between the index and thumb fingertips was used to implement the threshold, and a different value was applied for switching from each state to another state [71, 83]. Throughout the test, we selected an affordable distance for each threshold for the current study. As a result (Figure 3), the transition from semi-pinch to full-release pinch states was activated when the fingertip distance reached 10 cm, while a semi-pinch state was activated when the distance was between 2 to 7 cm. These thresholds allow users to perform the pinch gestures more clearly.

3.3 Subselection Triggering Methods

Next, we propose four subselection triggering methods. These were activated only when the DH was in the semi-pinch state. As in Figure 4, when an object is gazed at, the outline is highlighted in red (gray in user test) and changed to green (white in user test) if it is grouped. The green outline was disabled when it was ungrouped, and ungrouping individual objects proceeded in the same way.

3.3.1 Gaze Dwell. In this method, the gaze cursor had to intersect with the target collider and maintain the activated status (indicated in gray outline as in Figure 7 (A)) for 500 ms to group the target [14, 42, 62]. To prevent tracking issues caused by eye-tracking, these invisible colliders were set to have a three times bigger radius than the objects in this study [38]. Also, we tested several variations of gaze duration (250, 500, and 750 ms) and ended up using 500 ms as the best compromise between speed and error. This technique has the advantage of not requiring other hand gestures or movements, which could cause less arm fatigue than other techniques. Although the dwell method could be faster due to raid gaze input, it could potentially be even slower because of the high error rate due to the Midas touch problem reported in the previous studies [23, 42].

3.3.2 Swipe Left. Next, we propose a swipe method that uses hand movement to confirm grouping. Inspired by the action in smartphone lock screen (slide to unlock) [1], the user moves their DH toward the left in a semi-pinch state. In this technique, an indicator appears on the right side of the target to show the swipe status, such that users know how much they have to move to confirm grouping. As shown in Figure 7 (B), the indicator is designed with three spheres with the bigger ones located closer to the target. The indicator color was displayed white when the target object was in an ungrouped status and gray when it was in a grouped status, indicating the objects' post-activated status. The simple linear movement could enable fast and intuitive subselection. However, given the extensive potential range of hand movements, there is a possibility that they may exceed the intended range of motion.

When using movement-based methods, there was a minor issue of returning to the original position. Since hands can only move a certain distance, a return action is needed to enable a large number of subselections. Thus, we used the left direction of the hand movement and ignored the other side to prevent accidental subselection and make users' return action natural. For implementation, the movement vector of the DH was employed to move the indicator, which exhibited linear movement in a leftward direction from its initial position toward the target. The targets' grouping status was changed when the indicator touched the targets. On the other hand, the utilization of both hand movements on both sides (left and right) may facilitate area selection. This could be achieved by establishing an initial area position for the left swipe action and an end position with the right swipe action. However, this is not addressed in the current paper due to its scope.

3.3.3 Tilt Right. Similar to the swipe method, the tilt method also uses a hand movement on the semi-pinch hand. Similarly, as we rotate a knob to adjust the option (i.e., volume, channel) of a radio [51], users can tilt their hand (30°) to the right to confirm grouping. This technique also provides an indicator for tilt status. The tilt indicator consists of three spheres on the top of the target and has an additional sphere on the right side of the target that indicates the end of the rotation (Figure 7 (B)) as we rotate a knob to adjust the volume of radio [51]. As the swipe method indicator, the color of the tilt indicator was set to white if the target object was ungrouped and gray when it was in grouped status. In addition, only the right rotation vector was utilized, resulting in a solely rightward indicator movement. To map the hand rotation to the indicator, three times the acceleration was applied to the indicator rotation relative to the actual hand rotation. The target grouping was activated when the top spheres touched the end sphere. Tilt could keep the hand position more static compared to the swipe movement, but there could be potential tracking issues since the gap between fingertips for semi-pinch detection could be occluded during tilting. This false pinch could cause errors when performing a multi-selection (i.e., disable the grouping state by accidental single selection), which could result in increased completion time and fatigue.

3.3.4 NDH Pinch Clicking. Pinch clicking uses a full pinch gesture on the NDH to confirm the grouping while maintaining the DH semi-pinch. In this technique, the user maintains the gesture for switching modes and gazes at the target, then performs a full pinch

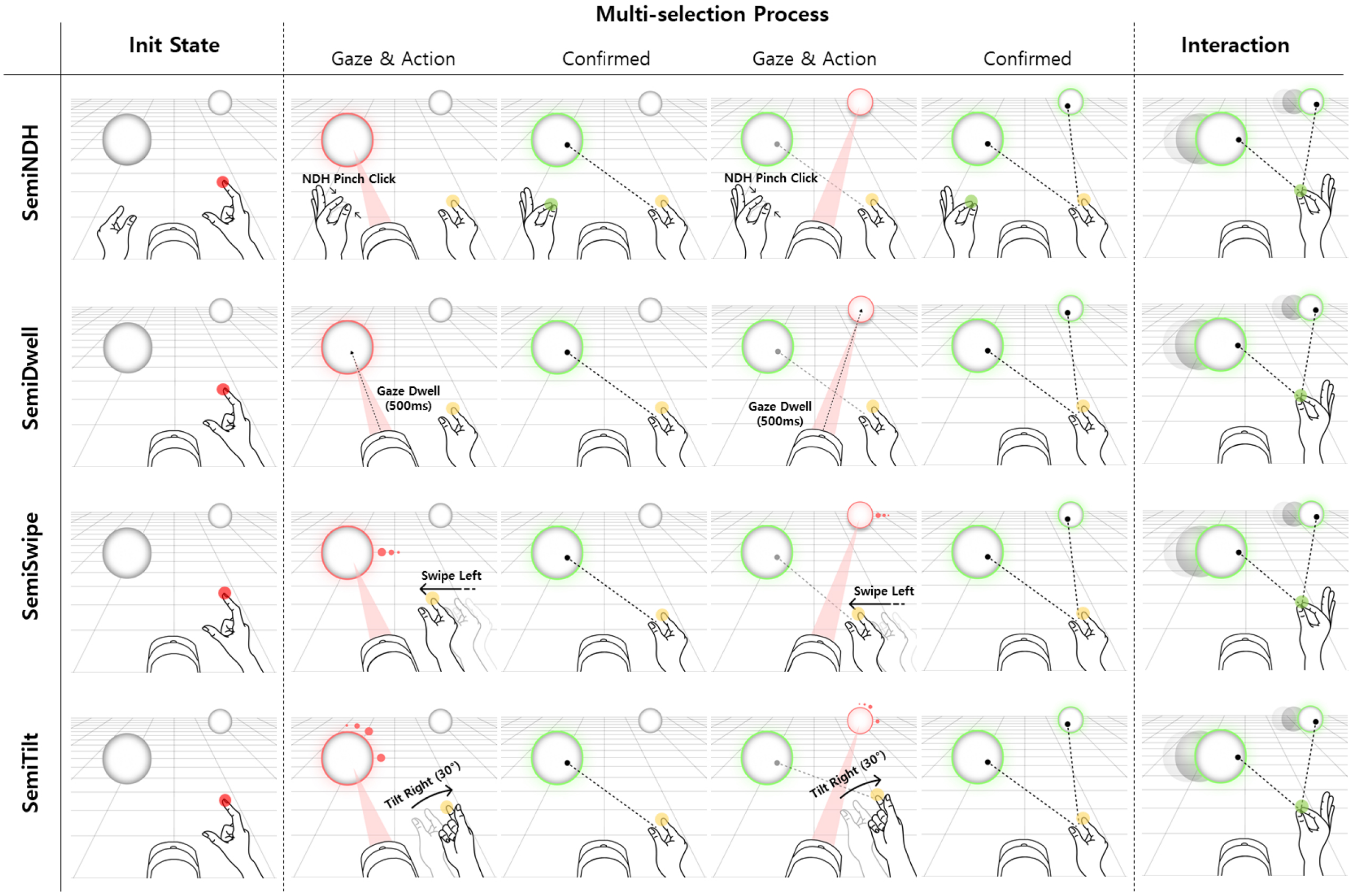

Figure 4: Illustration of PinchCatcher, serial multi-selection techniques utilizing the semi-pinch gesture for mode switching. While maintaining the semi-pinch gesture, the user performs a pinch-clicking motion with their non-dominant hand (SemiNDH), maintains gaze for 500 ms (SemiDwell), swipes left (SemiSwipe), or tilts right (SemiTilt) to confirm the grouping of the gazed object. A typical interaction flow involves four steps: (1) maintaining a semi-pinch to retain multi-selection mode, (2) directing gaze at a target and issuing a new command to sub-select it, (3) repeating step 2 until all targets are grouped, and (4) concluding with a full-pinch, which can then be used to drag all targets. After the interaction, users can disable the group by making a full-release pinch.

with the other hand to confirm the grouping of objects. The utilization of both hands with a semi-pinch gesture for operation is compatible with the default gesture set, as this particular gesture is not typically employed for other functions. We employ this method for comparisons with the single-handed mode switching techniques and with the movement-based subselection triggering techniques (i.e., swipe and tilt). This could offer advantages in terms of rapid pinch gestures but may require more attention on tracking both hand statuses for mode switching.

3.4 Interaction after Grouping

After grouping all targets, users can proceed to the next interaction by making a full pinch as in Figure 5. With the grouped objects, users can perform the same functions on multiple objects simultaneously, such as copy and paste, resize, dragging, etc [44, 47]. Once the interaction has been completed, users could ungroup the targets by making a full-release pinch or add more targets by making a semi-pinch gesture again.

4 EVALUATION

We conducted a user study to evaluate the usability and performance of each technique using a multi-selection task of targets on a 2D grid layout in a 3D environment [3, 37, 77]. Since we aimed to design a serial way of multi-selection, we employed the task that resembles selecting a UI component (i.e. photo, file), a common context for grouping under 10 target objects. We measured task completion time, error rate, and amount of hand movement, and collected subjective feedback.

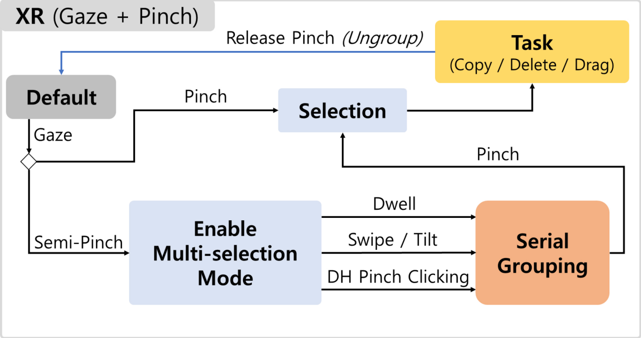

Figure 5: The multi-selection process with PinchCatcher.

4.1 Study Design

4.1.1 FullDH Technique (Baseline). We designed a mode-switching method using the NDH full-pinch for comparison with the one-handed semi-pinch method. This technique is based on the metaphor of a 2D desktop environment, pressing the CTRL key and clicking a mouse button for multi-selection [27, 74]. As in Figure 6, the user first makes a left-hand full pinch to activate the multi-selection mode. While maintaining a left-hand pinch, the user gazes at the target and pinches with their DH for each target serially to make a group. This technique could be intuitive and familiar to users, but it is not compatible with the default gesture interaction model [47], which uses DH pinch for selection. It requires users to switch the DH and NDH pinch depending on the selection mode.

We have also considered other baselines, such as area selection or UI buttons that enable multi-selection mode. However, these techniques were inappropriate because we were aiming for the serial way of multi-selecting, and buttons are not always accessible in the XR environment. In addition, hypothetically, we could have included a controller baseline, but it would have introduced potential confounds, such as varying tracking quality and haptic feedback, making the study more about a comparison between hand tracking vs. controllers. Thus, considering the scope of this study, we focused on essential factors of gestural design (i.e., uni vs. bimanual, dwell vs. motion vs. pinch) in the eye and hand-tracked XR UIs context. Therefore, we chose the FullDH technique as our baseline as it is similar to holding down the CTRL key in a familiar desktop environment.

4.1.2 Task Design. We designed a scattered target grouping task, employed in previous multi-selection studies [3, 37, 77]. The present study did not employ the task based on Fitts' law [14, 73], which is focused on evaluating selection interaction performance. This is because the objective of the present study is to evaluate the proposed techniques in a general context of XR applications and to collect data on overall interaction performance during multi-selection. However, the Fitts' law task focuses on the single selection task and is inappropriate for use in crowded or naturalistic settings.

A total of 40 sphere-shaped objects were arranged in equal intervals (1m) on a rectangular flat grid layout window (Figure 7 (A)) where the objects were randomly placed in each trial. The distance between the participant and the layout window was 13.5 m. The width of the window was set to 58.12° , and the height to 25.06°. The object radius was set to 0.2m each. Participants were asked to group all targets (blue) while avoiding distractors (white).

There were two independent variables in the current within-subject study. First, it consisted of five multi-selection techniques: FullDH, SemiNDH, SemiDwell, SemiSwipe, and SemiTilt. We selected FullDH as a baseline since it is based on the metaphor of a 2D desktop environment using the keyboard and mouse input for multi-selection, which is familiar to most participants. Second, there were three variations in the number of targets: 2, 4, and 6, representing a common range of number of targets for multi-selection [77]. We repeated each combination 15 times, resulting in 225 trials. In addition, we included two training trials before each block, consisting of 45 trials in which the grouping task was performed using the specific technique. During the training trial, only 10 spheres were presented in the layout, and participants could practice the technique they would use in the following block. The order of the blocks was counterbalanced using a Balanced Latin square.

4.1.3 Implementation. The multi-selection techniques and the study environment were implemented using Unity 3D Engine (2022.3.9f1) and deployed on a Meta Quest Pro (90 Hz, 111.24° FoV) using its embedded eye tracker (30 Hz). We used Meta XR-all-in-One SDK's support for gaze and hand tracking. Due to the minor gaze tracking error of the current hardware system [19, 61, 65] and layout of 40 small objects, we modified the radius of the invisible object collider, which determines whether the user is gazing at an object, as the previous gaze-based interaction research [38]. We tried to exclude this error to collect data that focused on the multi-selection interaction. Accordingly, the collider was set to three times the radius (60 cm) of the visible object (20 cm). Similar to multi-selection on touchscreen (i.e., touch dwell before dragging grouped files) [12], we applied pinch-dwell (250 ms) to prevent errors caused by unintended pinch detection due to tracking issues or accidents.

4.2 Procedure

The experimenter asked participants to complete the consent and demographics forms and briefed them about the task and each multi-selection technique. Then, participants wore the HMD and performed eye-tracking calibration. Before starting each block, two training phases were provided to familiarize the participants with the techniques. Participants spent an average of 24.4 seconds (SD=1.71) and performed 10.77 (SD=3.73) multi-selection in each training phase. During the trial, participants were asked to group all the targets using the assigned technique as fast as possible (Figure 8). Participants performed and maintained a full pinch for 250 ms after grouping all the targets to complete the trial. After each trial, an inter-trial interval (ITI) scene showing a cross in the center instead of object layouts was presented, and participants were asked to make a full-release pinch gesture (Figure 4) to proceed to the next trial. After finishing the tasks using each technique, participants were asked to complete a questionnaire about their experience with the multi-selection technique and rest until they were ready to proceed to the next block. After finishing all five blocks, we asked participants to rank the techniques and conducted a brief interview on the ranking. The experiment lasted for 50 minutes on average.

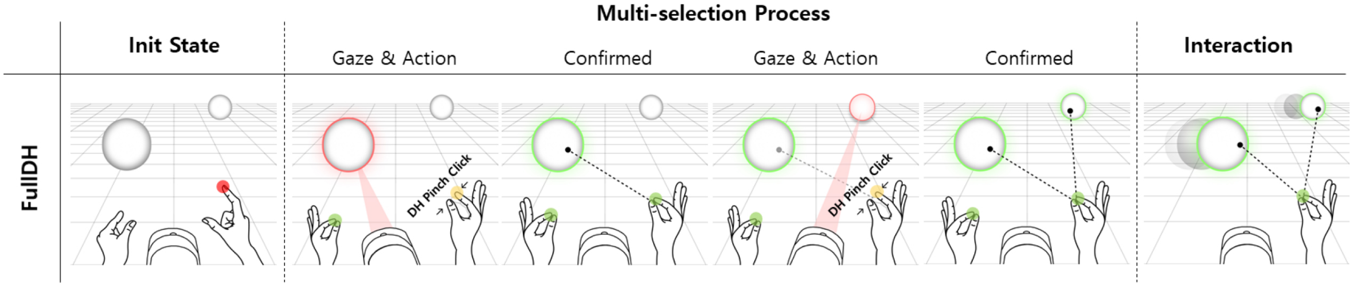

Figure 6: Illustration of FullDH technique process. We included this technique as a baseline, which resembles the mechanism of 2D desktop multi-selection. The FullDH uses a non-dominant hand (NDH) full pinch to activate a multi-selection state instead of a semi-pinch. While maintaining the full pinch gesture, users perform pinch-clicking with their dominant hand (DH) to subselect the gazed object.

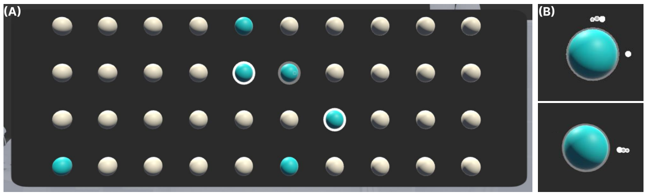

Figure 7: The overall experimental environment and stimulus design. (A) A flat grid layout for object display. A total of 40 objects were placed on the layout, with an equal interval spacing between objects. If the participants gazed at the object, the outline was highlighted in gray. If it was in a grouped state, the object outline was set to white until it was ungrouped. (B) The indicator design for the SemiTilt (top) and SemiSwipe (bottom) techniques, respectively.

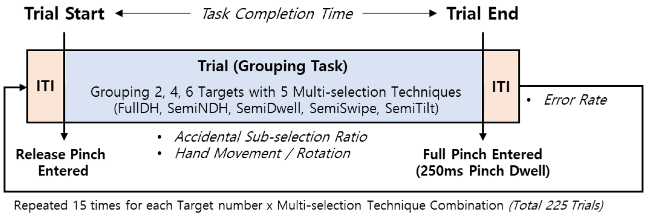
Trial Start
Task Completion Time
Trial End
ITI
Trial (Grouping Task)
Grouping 2, 4, 6 Targets with 5 Multi-selection Techniques
(FullDH, SemiNDH, SemiDwell, SemiSwipe, SemiTilt)
ITI
• Error Rate
Release Pinch Entered
• Accidental Sub-selection Ratio
• Hand Movement / Rotation
Full Pinch Entered (250ms Pinch Dwell)
Repeated 15 times for each Target number x Multi-selection Technique Combination (Total 225 Trials)

Figure 8: The illustration of the trial procedure. Inter-trial interval (ITI) was provided before and after each trial. The trial started with a release pinch in the ITI state and ended when participants made a full pinch for 250 ms after grouping targets. During the trial, task completion time, accidental subselection case, and hand movement and rotation were collected. After each trial, the error rate was measured by analyzing the grouping result (i.e., the number of targets that failed to be grouped and distractors in the final group).

4.3 Evaluation Metrics

- *Task Completion Time (TCT)*: The interval between the moment when the targets appeared to the moment when a full pinch was performed to end the trial.
- *Accidental Subselection Ratio (During Trial)*: The ratio was calculated as the number of distractors (white spheres) grouped accidentally during each trial divided by the total number of subselections performed per trial.
- *Error Rate (After Trial)*: The sum of errors after each trial (the number of targets that failed to be grouped by participants, and the number of distractors that were in the final group) divided by the total number of grouped objects per trial.
- *Inverse Efficiency (After Trial)*: The grouping success rate was calculated as the percentage of trials with no errors (distractor included or target missed to group) in the final group. Inverse efficiency (IE) was calculated by dividing the TCT by the grouping success rate and it indicates the combined effects on grouping efficiency [63, 68]. Therefore, high IE values indicate low grouping efficiency of techniques.
- *Hand Movement and Rotation*: The hand movement and rotation while performing multi-selection during each trial were measured. As in previous studies [3, 73], we used the center of the palm translation and rotation value of each frame.
- *Questionnaire*: We measured the experience using the System Usability Scale (SUS) [6], NASA Task Load Index (NASA-TLX) [15], and a single question about satisfaction [79] on a 7-point Likert scale (0 to 6). The SUS is comprised of three positive and negative statements, with the score for the negative statement subtracted from six and subsequently summed with the positive statement scores. The NASA-TLX has six features including mental, physical, and temporal demand, as well as perceived performance, effort, and frustration.
- *Ranking & Brief Interview*: At the end of the study, participants were asked to rank the five techniques in a survey. Open interviews were also conducted regarding the reason for their ranking and their experience with each technique.

4.4 Participants

We recruited 30 participants (Mean age=25.93, SD=4.63, 15 Male) from the local university. Among the participants, 29 were right-handed, 8 wore glasses, 12 wore contact lenses, and 7 had vision correction surgery. We asked participants about their prior experience of using VR, controller, hand, and eye interaction on a 6-point Likert scale (0-5 points). The participants responded to their VR experience with an average rating of 2.76 (SD=1.41). For interaction modalities, they rated their experience with VR controllers at 2.83 (SD=1.21), with hands at 2.21 (SD=1.61), and with gaze at 2.07 (SD=1.46). All study protocols and methods were approved by the university's Institutional Review Board (IRB), and all participants were rewarded $10 for participating in the experiment.

5 RESULT

A total of 28385 multi-selections including accidental subselections were collected and it was analyzed using two-way repeated measures ANOVA with five multi-selection techniques (FullDH, SemiNDH, SemiDwell, SemiSwipe, SemiTilt) and three target numbers (2, 4, and 6) as within-subject factors with Bonferroni correction. The Wilcoxon Signed-Rank Tests were used for post-hoc on the distractor-grouped case and error rate metric, due to violations of the normality assumption. For the nonparametric questionnaire data, we used the Friedman test and Wilcoxon signed rank tests for post-hoc analysis. In terms of TCT, hand movement, and rotation analysis, we excluded trials in which participants accidentally skipped or failed to complete the task. Given that the variation in target numbers was even and that the minimum target was two, we filtered out trials where participants failed to group over 50% of the total targets, resulting in the exclusion of 30 trials (0.44%).

5.1 Task Completion Time (Figure 9)

As the number of targets increased (F_c(1.747, 50.673)=725.511, p<.001, $\eta_p^2 = .962$), participants took more time to complete the task (2 targets (M=4381.942, SD=1293.517) < 4 targets (M=6691.875, SD=2107.008) < 6 targets (M=9029.705, SD=2359.307)) (t(29)s>19.244, ps<.001). However, there was no main effect of multi-selection techniques (F_c(2.705, 78.457)=1.028, p=.379, $\eta_p^2 = .034$) or the interaction effect (F_c(5.274, 152.945)=.898, p=.488, $\eta_p^2 = .030$).

We performed the TCT analysis with error-free data by excluding 445 trials (6.5%). Overall, the test showed similar results, indicating a main effect on target number (F_c(1.662, 48.192)=749.745, p<.001, $\eta_p^2 = .963$). As the number of targets increased, participants spent more time completing the task (2 targets (M=4382.346, SD=1312.669) < 4 targets (M=6656.641, SD=1989.346) < 6 targets (M=9044.002, SD=2415.898)) (t(29)s>20.398, ps<.001). There was no main effect of techniques (F_c(2.754, 79.873)=1.276, p=.288, $\eta_p^2 = .042$) or interaction effect (F_c(4.794, 139.036)=.846, p=.516, $\eta_p^2 = .028$).

5.2 Accidental Subselection Ratio–During Trial (Figure 10 (A))

The repeated measures ANOVA and post-hoc Wilcoxon signed-rank tests on multi-selection technique (F_c(3.126, 90.641)=9.550, p<.001, $\eta_p^2 = .248$) indicated that the FullDH (M=8.650, SD=7.633) technique induced a significantly higher accidental subselection ratio than other techniques (Zs>2.746, ps<.006): SemiNDH (M=5.163, SD=4.984), SemiDwell (M=4.206, SD=5.032), SemiTilt (M=3.380, SD=3.177), and SemiSwipe (M=2.436, SD=3.786). In addition, the result showed that the SemiSwipe technique induced significantly less accidental subselection ratio than SemiDwell (Z=2.067, p=.039) and SemiNDH (Z=2.552, p=.011). The main effect of target number (F_c(1.740, 50.469)=3.150, p=.058, $\eta_p^2 = .098$) and the interaction effect (F_c(4.980, 144.427)=.760, p=.580, $\eta_p^2 = .026$) were not significant.

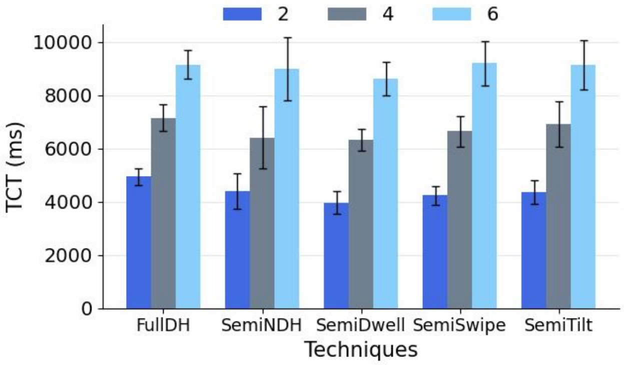

Figure 9: The average task completion time (TCT) in each target number condition and multi-selection technique. Whereas larger target numbers led to longer TCT, no differences were found between the techniques. The error bars represent 95% confidence intervals.

5.3 Error Rate–After Trial (Figure 10 (B))

For the main effect of the multi-selection technique on error rate (F_c(2.208, 64.023)=19.177, p<.001, $\eta_p^2 = .398$), SemiSwipe (M=2.767, SD=5.928) showed a significantly lower error rate than SemiDwell (M=8.612, SD=11.483) (Z=4.088, p<.001) and SemiTilt (M=11.648, SD=15.328) (Z=4.165, p<.001), but a higher error rate than FullDH (M=1.207, SD=3.599) (Z=2.974, p=.003). Also, it showed that the SemiDwell and SemiTilt techniques had significantly higher error rates than FullDH (SemiDwell: Z=4.704, p<.001 / SemiTilt: Z=4.535, p<.001) and SemiNDH (M=1.854, SD=6.259) (SemiDwell: Z=4.022, p<.001 / SemiTilt: Z=4.371, p<.001), respectively. There was no main effect of target number (F_c(1.756, 50.912)=1.710, p=.194, $\eta_p^2 = .056$) or interaction effect (F_c(3.211, 93.112)=1.996, p=.116, $\eta_p^2 = .064$).

5.4 Inverse Efficiency–After Trial (Figure 10 (C))

The IE data showed a significant main effect of the multi-selection technique (F_c(2.606, 75.570)=4.379, p<.001, $\eta_p^2 = .131$). The post-hoc tests showed that the mean IE of the SemiNDH (M=6719.815, SD=1186.851) was significantly lower than the SemiTilt (M=8994.113, SD=3095.398) technique, indicating that the SemiNDH has higher grouping efficiency than SemiTilt. There was no significant difference in efficiency between the other techniques. There was also a significant main effect of target number (F_c(1.775, 51.469)=274.505, p=.009, $\eta_p^2 = .904$), with the IE increasing as the target number increased (2 targets (M=4795.791, SD=1630.533) < 4 targets (M=7556.147, SD=3107.372) < 6 targets (M=10495.514, SD=3784.793)). Lastly, we found an interaction effect between the technique and target number factor (F_c(4.916, 142.551)=3.422, p=.006, $\eta_p^2 = .106$). The post-hoc tests showed that the benefit in efficiency for semiNDH over semiTilt was significant starting from grouping 4 target objects (Δ=2243.428, t(29)=3.408, p=.002) and the size of the benefit

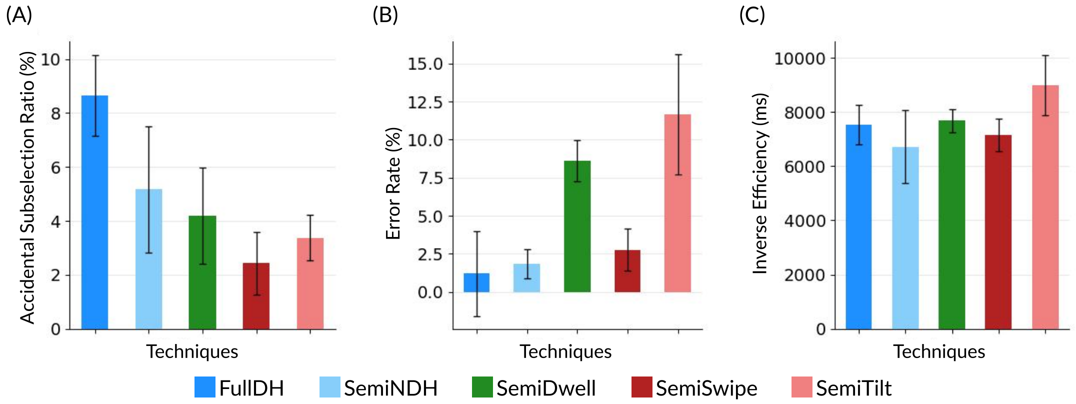

Figure 10: (A) Accidental Subselection Ratio: how often distractors were accidentally subselected out of all subselections during a trial. FullDH showed significantly higher accidental subselection than other techniques. (B) Error Rate: total errors including ungrouped targets and mis-grouped distractors, divided by the total number of grouped objects per trial. SemiDwell and SemiSwipe showed significantly higher error rates than other techniques, but not between them. (C) Inverse Efficiency (IE): task completion time divided by the percentage of error-free trials, with a lower IE indicating better grouping efficiency. SemiNDH was significantly more efficient than SemiTilt. Error bars represent the 95% confidence intervals.

became even greater when grouping 6 target objects (Δ=3583.793, t(29)=4.686, p<.001).

5.5 Hand Movement (Figure 11 (A))

There was a main effect of the multi-selection technique (F_c(1.339, 38.827)=128.688, p<.001, η_p^2 = .816) showing that the hand movement of each technique was significantly different from each other (t(29)s>5.542, ps<.001), except between SemiNDH and SemiDwell (t(29)=.677, p=.504). This result indicates that participants had to move their hand most in the SemiSwipe technique (M=1.470, SD=.864), followed by SemiTilt (M=.907, SD=.443) and FullDH (M=.275, SD=.158). Participants moved their hands least when using the SemiNDH (M=.152, SD=.085) and SemiDwell (M=.143, SD=.059) techniques. In addition, there was also a significant main effect of target number (F_c(1.268, 36.780)=261.889, p<.001, η_p^2 = .900). Participants moved their hands more as the target number increased (2 targets (M=.354, SD=.333) < 4 targets (M=.593, SD=.628) < 6 targets (M=.822, SD=.892)) (t(29)s>14.034, ps<.001). In addition, there was also an interaction effect between the multi-selection technique and target number (F_c(2.060, 59.745)=99.846, p<.001, η_p^2 = .775). The post-hoc results demonstrated that all techniques exhibited significant differences in all target number cases (t(29)s>4.237, ps<.001), except for the comparison between SemiNDH and SemiDwell in all target number cases (t(29)s<.938, ps>.356). These results suggest that all techniques demanded significantly more hand movement as the target number increased. However, the SemiSwipe, SemiTilt, and FullDH techniques exhibited markedly more extreme increases as the target number increased than the other techniques.

5.6 Hand Rotation (Figure 11 (B))

We found a main effect of the technique (F_c(1.352, 39.199)=155.639, p<.001, η_p^2 = .843) indicating that the amount of hand rotation performed during the trial was significantly different from each other (t(29)s>7.833, ps<.001), except between SemiNDH and SemiDwell (t(29)=1.652, p=.109). The SemiTilt technique (M=687.451, SD=389.389) made participants rotate their hands most, and then SemiSwipe (M=219.246, SD=144.122) and FullDH (M=63.040, SD=31.155) followed. In the SemiNDH (M=26.903, SD=16.990) and SemiDwell (M=23.167, SD=10.555) technique conditions, the participants demonstrated the lowest level of hand rotation. There was also a main effect of target number (F_c(1.404, 40.714)=211.932, p<.001, η_p^2 = .880) indicating that participants rotated their hands more as the target number increased (2 targets (M=155.715, SD=145.56) < 4 targets (M=208.130, SD=301.49) < 6 targets (M=288.039, SD=410.79)) (t(29)s>11.225, ps<.001). There was also an interaction effect (F_c(1.870, 54.234)=113.369, p<.001, η_p^2 = .796) between the multi-selection technique and the number of targets. It indicated that participants rotated their hands more as the target number increased using the SemiTilt, SemiSwipe, and FullDH techniques (t(29)s>6.743, ps<.001). However, there was no significant difference in hand rotation when using the SemiDwell (Δ=2.83, t(29)=1.985, p=.057) and SemiNDH (Δ=.329, t(29)=.312, p=.758) techniques between 4 and 6 targets. SemiDwell only showed a significant difference between 2 and 6 targets (Δ=4.184, t(29)=3.718, p<.001), and SemiNDH showed a significant difference between 2 and 4 targets (Δ=3.721, t(29)=3.553, p<.001) and 2 and 6 targets (Δ=4.050, t(29)=3.689, p<.001).

5.7 Questionnaire (Figure 12 (A))

There was a main effect of the multi-selection technique only on physical demand (χ^2(4)=15.909, p=.003) and performance (χ^2(4)=20.358, p<.001) for NASA-TLX. The post-hoc result showed that SemiNDH required more physical demand than SemiDwell (Z=3.666, p<.001) and FullDH (Z=2.376, p=.017). Also, the score of SemiTilt was higher than SemiDwell (Z=2.902, p=.004) for physical demand. In terms of perceived performance, both FullDH and SemiNDH demonstrated a higher score than SemiTilt (FullDH: Z=2.757, p=.006 / SemiNDH: Z=2.421, p=.015) and SemiDwell (FullDH: Z=3.111, p=.002 / SemiNDH: Z=1.975, p=.048), respectively. Also, SemiSwipe showed

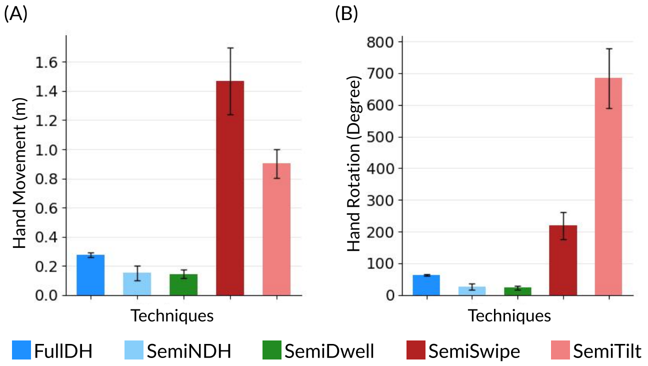

Figure 11: Average (A) hand movement and (B) hand rotation during the trial in each multi-selection technique. All differences between multi-selection techniques were significant except between SemiNDH and SemiDwell in both features, showing they resulted in the least physical hand motion. The error bars represent 95% confidence intervals.

better performance than SemiTilt (Z=2.445, p=.014). Moreover, the perceived usability (SUS) scores of all five techniques averaged over 4.05 points, which is the SUS criterion for minimum usability [54].

5.8 Ranking & Qualitative Feedback

Participants responded that they liked the SemiDwell (9) and SemiSwipe (8) techniques the most. However, SemiDwell was also rated as the least preferred technique (10), indicating extreme individual differences in preference. Additionally, 11 participants rated FullDH in second place, while SemiNDH was ranked in fourth place by 11 participants. This shows that most participants also preferred the FullDH but not the SemiNDH technique (Figure 12 (B)) The detailed rationale for each ranking can be found in the following.

5.8.1 FullDH. The majority of participants indicated that the FullDH technique was intuitive and familiar (P7, P19, P20) and fast (P6, P21). P24 mentioned that the sensation of the finger touch when pinching for subselection provided haptic feedback. However, they also indicated that maintaining both hands within the tracking space resulted in greater fatigue than the one-handed multi-selection technique (P3, P13, P22, P29), and it was confusing to keep tracking the states of both hands (P1, P2, P5, P13). Some participants mentioned that since two different functions (group confirm, trial end) were assigned on the right-hand pinch, it occasionally led to errors (P11, P30) due to confusion.

5.8.2 SemiNDH. Similar to the FullDH technique, most participants mentioned that the SemiNDH was straightforward (P11, P13, P21) but also caused arm fatigue due to using both hands (P5, P16). Some participants indicated that maintaining the right-hand semi-pinch state required more attention than maintaining the left-hand full pinch in the FullDH technique (P6, P23). Additionally, several participants reported that pinch-clicking with the NDH felt awkward (P15, P24). For instance, P15 mentioned, "I am right-handed, so it felt awkward to pinch with my left hand. I prioritized pinching with my right hand a little higher." In addition, P2 also stated, "It was confusing because I was using my left hand for subselection and then suddenly tried to finish it with my right hand. It felt like I was going back and forth between the two hands."

5.8.3 SemiDwell. Participants who preferred the SemiDwell technique mentioned that the absence of hand movement for subselection and one-handed interaction made them physically comfortable (P5, P7, P10, P12, P17). Nevertheless, some have commented that they experienced a lack of stability due to the objects being frequently grouped unintentionally (P5, P8, P9, P11, P27) and experienced significant eye fatigue from repeated usage (P20, P22). Similarly, P4 expressed frustration when attempting to accelerate grouping, given the fixed dwell time of 500 ms.

5.8.4 SemiSwipe. Participants indicated that one-handed interaction increased their comfort level (P2, P6, P8, P16). Moreover, participants indicated that SemiSwipe was the most comfortable and stable of the one-handed techniques (P6, P23, P27, P28, P29). They mentioned that SemiSwipe was more stable and faster for multi-selection due to its gaze independence (P2, P10, P16, P28). However, participants indicated they needed time to familiarize themselves with the technique (P24, P25). In addition, they mentioned the need to consider the range of action and how the hand returns (P5, P11). They also reported experiencing physical fatigue due to the additional action required compared to other techniques.

5.8.5 SemiTilt. Regarding the SemiTilt technique, participants mentioned they could group faster using a simple one-handed movement (P2, P22, P27). However, similar to the SemiSwipe technique, participants also commented on physical fatigue, with more tension required on the wrist and shoulder in the SemiTilt technique (P5, P6, P14, P17, P19, P23). They also considered the action of returning to the initial hand position ambiguous (P11, P13), and SemiTilt more sensitive to hand movement (P9, P25, P28, P30).

6 DISCUSSION

6.1 Semi-pinch based One-handed Multi-selection Interaction

Regarding the FullDH technique, the keyboard-mouse metaphor would have been beneficial because of the familiarity with the habit and motor concept of using gaze along with DH. We found this aspect from the interview and questionnaire results. In contrast, the pinch-clicking action with the NDH in the SemiNDH technique made participants feel more confused and awkward. This finding aligns with the bimanual interaction principles proposed by Guiard [13] that guide the utilization of NDH for coarse, irregular action and DH for detailed and frequent action to activate the designated function. Thus, participants found the FullDH technique, which assigned mode switching to NDH pinch and rapid pinch-clicking for subselection to DH, more familiar and straightforward than the SemiNDH technique.

However, there were limitations regarding the coupling between gaze and pinch gestures. For instance, the familiar and fast mechanism tended to induce more accidental subselection during grouping, and participants had to keep track of both hand states while performing multi-selection. These limitations caused too many errors during the trial, which had to be corrected afterward. Eventually, the bimanual techniques could perform fast subselection but

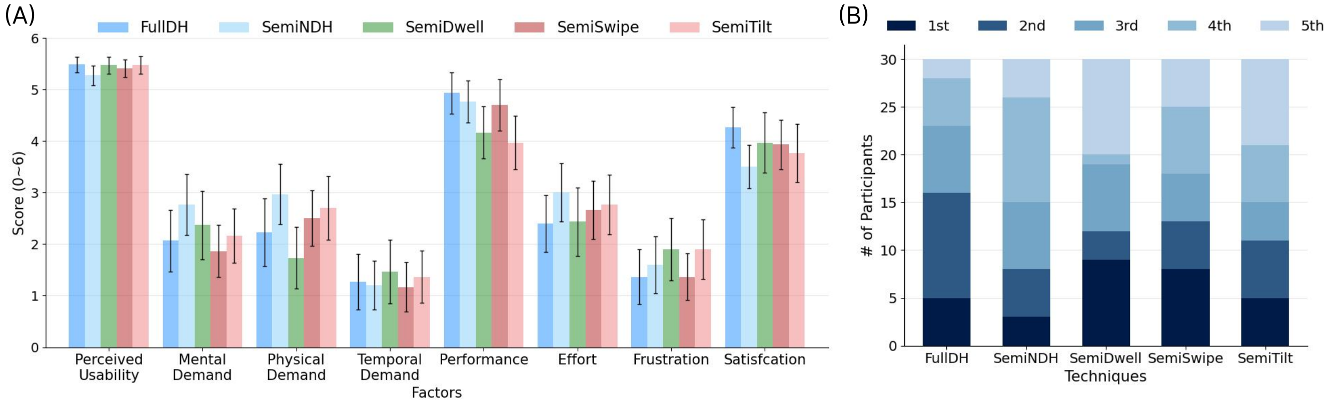

Figure 12: The result of quantitative measures on the experience of using each multi-selection technique. (A) The average score of each factor in the questionnaire (i.e., SUS, NASA-TLX, Satisfaction) on a 7-point Likert scale (0-6). The statistical results indicate that there was a significant main effect of the multi-selection technique on physical demand and performance factors. Also, all techniques have received an average SUS score over 4.05, which is the criterion for minimum usability [54]. The error bars represent the 95% confidence intervals. (B) The result of the participants' ranking of each technique.

resulted in TCT similar to that of other one-handed techniques. Still, as the previous work of bimanual interaction indicates that the performance could differ depending on the tasks [17, 38], the FullDH technique could be affordable outside of contexts that require quick and accurate subselection.

Overall, the semi-pinch-based one-handed techniques showed better performance in error during subselection and user evaluation, especially when a swipe was used for subselection. Participants preferred the one-handed technique since it required less attention on the hand gesture status and enabled them to focus solely on the grouping task. Also, the semi-pinch provided an intuitive and straightforward interaction process for participants, such that they could adapt quickly. However, there are also limitations in semi-pinch-based techniques, such as the varying performance depending on the subselection triggering method.

With the one-handed technique, users can perform other tasks (e.g., moving objects before sub-selecting for alignment, resolving occlusion, scrolling down lists) that could support multi-selection, whereas the bimanual technique requires all resources for one task. For instance, users can remove an occluding object with their NDH and then perform multi-selection with their DH. Therefore, the one-handed multi-selection technique has the potential to enhance compatibility with other pinch-based interactions, not only interaction after grouping but also throughout the multi-selection process.

6.2 Hand Movement for Triggering Subselection

Compared to the pinch clicking and dwell method for subselection, we found that the method with hand swiping outperformed other methods in terms of error and usability. Participants also mentioned that gaze dwell was comfortable since it required no extra hand movement. However, the error metric results indicated that the Midas touch problem limited the gaze-dependent technique (SemiDwell). Thus, to adopt the dwell-based technique in the application, appropriate dwell time lengths need to be adjusted for each user with the value that they can tolerate and handle the errors.

Pinch-clicking-based methods (i.e., FullDH and SemiNDH) caused high errors due to gaze-independent aspects (i.e., no visual indicators and sudden/rapid subselection); in contrast to the hand movement-based techniques, the visual features, such as an indicator of gaze and hand status, are less involved in the subselection process. The pinch action could be sudden and fast, making it difficult for users to track their selection status while sub-selecting. As a result, users could occasionally make the pinch gesture before their gaze touches the object they want to sub-select and make an early trigger error [32]. Users could redo the selection if they selected the wrong object in a single selection task, but it could be problematic in a multiple selection task that requires users to correct their mistakes additionally.

The current study utilized micro-gestures such as hand swipe and tilt for the subselection method. However, the compatibility of the micro-gesture with gaze must be considered when designing a subselection triggering method [46]. We found that the visual indicator plays an important role in enhancing this compatibility. For instance, the swipe method showed an overall low error rate and comparable performance to the pinch clicking-based techniques in the NASA-TLX outcome. The linear swipe and 1:1 indicator to hand movements were intuitive and worked positively for the participants. Thus, the results of this study indicated that SemiSwipe is the most appropriate technique for multi-selection in an eye-hand interaction context among the designed techniques.

On the contrary, participants did not prefer the SemiTilt technique in this study, not only because it caused more arm fatigue but also because of the high sensitivity of indicator movement to hand movement. We tried to prevent potential tracking issues, such as false pinch detection when rotating DH, by employing the threshold of 30° instead of 90°. Still, several participants experienced tracking errors during the trial and felt it was too sensitive. This could depend on the participants' pinch style (i.e., preferred pinch gesture direction) and hand size. The incorporation of sophisticated hand-tracking capabilities for micro-gestures [31, 41] and individual hand calibration for personalized semi-pinch detection threshold [83], have the potential to enhance the precision and usability of SemiTilt.

7 APPLICATIONS

Based on the study results, we illustrate two scenarios in which multi-selection could be generally used in XR applications: File management and RTS Game. By presenting the application scenarios, we aim to provide direct insight and design guidelines for adopting multi-selection functions. All five techniques are presented in each application scenario with the same interaction process as the user test. However, we applied additional visual representation for grouping status by adding an individual line connecting grouped objects to the right-hand fingertip for clearer indication.

7.1 2D Environment Context: File Management

First, we present the file management application, which represents a fundamental task in the 2D environment for various domains (i.e., programming, design, office work) [10, 27, 37]. Since file management mostly consists of sub-tasks such as dragging files to other folders or applying specific operations (i.e., deleting, duplicating, copying, compressing) on multiple files, a multi-selection function would be more useful. Furthermore, providing an efficient method for managing multiple objects in XR becomes more crucial, as recent XR is capable of presenting multiple windows at once [9, 35].

As in Figure 13, we implemented a photo management application with sharing, moving, deleting, and copying functions. Users can select a photo that they want to manage by gazing at it and making a semi-pinch to select multiple photos using the PinchCatcher techniques. After selecting the photos, users can share them with other people by dragging them to the target avatar. They can also move the photos to other folder windows in the same way. Next, when users hold a full pinch with the photos selected for 2s, the button interface for the delete and copy option is displayed. Then they can select the deleting and copying function by gazing at the button and releasing a pinch to activate the function.

7.2 3D Environment Context: RTS Game

The XR environment consists of more complex features and contexts than the 2D environment, which makes it difficult to respond to the previous multi-selection techniques that require multiple steps for mode switching. However, we have found that our multi-selection design allows for rapid grouping and intuitive mode switching based on user test results. As a result, our techniques are adaptable and responsive to more complex and dynamically changing situations. Additionally, the multi-selection function is capable of maintaining alignment between objects during dragging [37, 58]. This is an advantage in contexts where the 3D models need to be moved in formation, such as furniture placement or real-time strategy (RTS) gaming. Therefore, we present RTS gaming as an example of a 3D environment context that could represent both complex context response and alignment features of multi-selection.

As in Figure 14, we implemented an RTS game in which players have to protect their planets from the invasion of enemy fighters. As in the file management application, the players can select the fighters by gaze and pinch interaction, and perform multi-select with the designed techniques. When the fighter is placed close enough to the enemy, it begins to engage. With the proposed multi-selection technique, players could quickly and appropriately distribute their fighters based on their strategy and also maintain the alignment to attack the enemy in a more efficient formation.

8 LIMITATIONS & FUTURE WORK

The techniques featured in the current study were all designed to perform serial multi-selection tasks, without covering parallel techniques for scenarios such as selecting a large number of targets that are densely arranged [56, 77]. While we believe that semi-pinch-based gaze selection techniques are most suitable for serial multi-selection tasks that cover the most common scenarios in XR, we concur that parallel multi-selection techniques also need to be investigated in the future. One potential parallel technique based on our proposed methods is using a swipe movement while maintaining a semi-pinch. This technique could allow the user to define two points by swiping left and right, respectively, in the 3D space for area selection.

Since we focused on the Gaze + Pinch interaction context, we compared hand-related features (i.e., unimanual vs. bimanual and dwell vs. motion vs. pinch) within the interaction. However, comparison with other input modalities, such as ray-casting with a controller, could provide additional insights into usability. In addition, the efficiency of each confirmation method could be explored more in the Fitts' law-based task. These were not explored in this study, but these features could strengthen the insight for multi-selection interaction in XR and should be explored in the future.

There were also some restrictions in the implementation variables, such as the collider size for target selection and the threshold distance for pinch state. For the proposed techniques to be universally applicable, it would be essential for these values to be adjustable depending on the context and individual characteristics. Personalization approaches based on the hand size or exploring the usage of left-handed users could help reduce the cognitive load of using semi-pinch gestures.

The current study presents the objects on a 2D flat layout with different colors that differentiate between targets and distractors. However, in most actual applications, targets are not highlighted, necessitating users to search and classify the objects before grouping them. For example, when users want to remove files from the list, they must consider the file features (e.g., name, size, date, etc.) to select the files that need to be removed. Also, there are diverse features in the XR environment, such as depth and dynamic targets, that could affect the performance. In these contexts, gaze-dependent techniques such as SemiDwell may result in more errors due to the Midas Touch problem [23, 42]. Consequently, further study is required to explore multi-selection techniques in broader contexts that consider 3D features to suggest appropriate techniques for future XR applications.

Although the SemiSwipe technique was preferred and showed better performance than other techniques, participants tended to swipe more than the activation threshold. The SemiTilt also had a similar but more critical issue, resulting in poor performance and evaluation. These limitations could be improved with the advanced algorithm or additional tracking sensors. As our study demonstrated

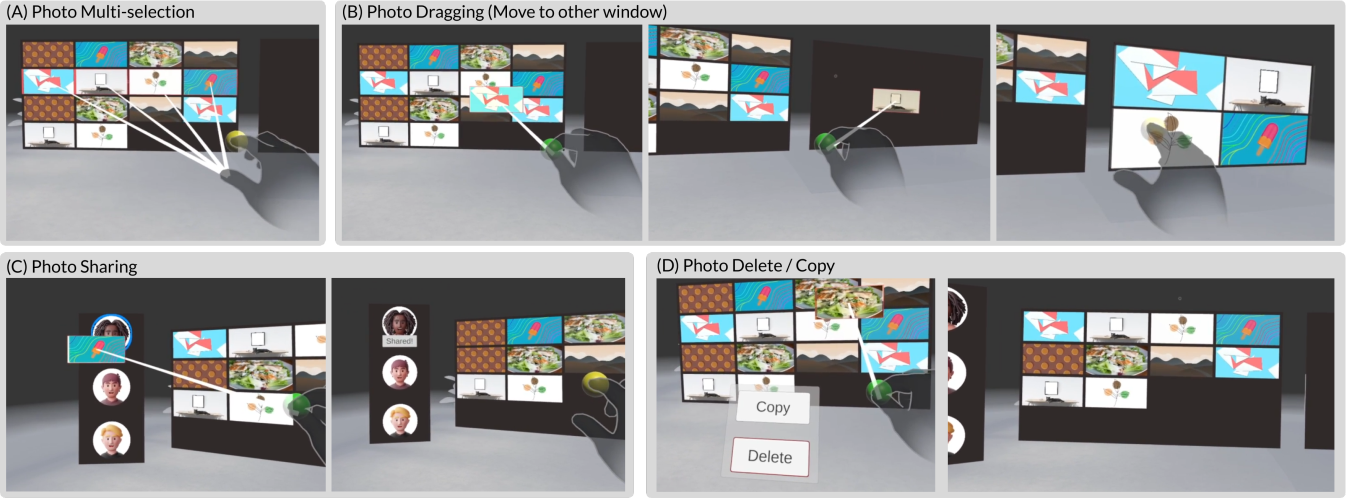
(A) Photo Multi-selection
(B) Photo Dragging (Move to other window)
(C) Photo Sharing
Shared!
(D) Photo Delete / Copy
Copy
Delete

Figure 13: The file management application: (A) Users can group multiple photos with the PinchCatcher techniques. (B) Drag and relocate photos to other windows/folders. (C) Share photos with other users. (D) Selecting an option for copying or deleting the grouped photos with gaze and full-release pinch. After the function was activated, the grouped photos were deleted.

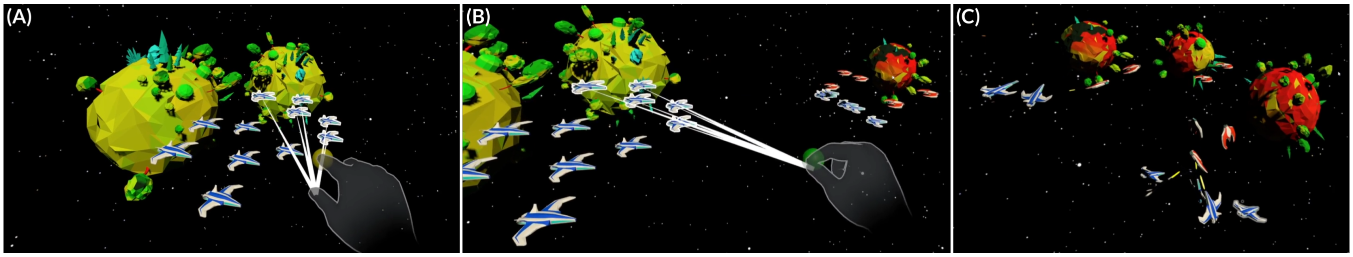

Figure 14: The RTS game application: (A) Grouping fighters by using the SemiSwipe technique, (B) Dragging multiple fighters to appropriate locations based on their strategy, (C) Start combat with the enemy fighters.

that visual feedback is a key factor in the compatibility of hand movement with gaze, it may also have the potential to improve these limitations. Therefore, future research could explore various forms of indicators and feedback that could indicate the subselection status and prevent excessive movement. For instance, auditory or tactile feedback could be used with visual feedback to enhance the awareness of users in tracking the subselection status [26].

9 CONCLUSION

This paper proposes the PinchCatcher, techniques for multi-selection function for eye-hand interaction in the XR environment. We designed one-handed multi-selection techniques using a semi-pinch hand gesture for mode switching while using microgestures for subselection (i.e., SemiSwipe, SemiTilt). In the user study, we compared these techniques with the bimanual dominant-hand full pinch method (i.e., FullDH) and non-movement-based subselection triggering methods (i.e., SemiNDH, SemiDwell). As a result, one-handed interaction with semi-pinch mode switching and swipe hand movement for subselection (SemiSwipe) showed the least errors, received higher scores on the evaluation, and was most preferred by the participants. Based on these findings, we presented application scenarios where our techniques could improve the user experience of multi-selection tasks in XR environments.

ACKNOWLEDGMENTS

This research was supported by the National Research Council of Science & Technology (NST) grant by the Korea government (MSIT) (No. CRC21014), and the STEAM R&D Project, National Research Foundation (NRF), Republic of Korea (RS-2024-00454458).

REFERENCES

[1] Ahmed Sabbir Arif and Ali Mazalek. 2014. Slide-to-unlock revisited: two new user authentication techniques for touchscreen-based smartphones. In *Proceedings of the 11th International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services* (London, United Kingdom) *(MOBIQUITOUS '14)*. ICST, Brussels, BEL, 389–390. https://doi.org/10.4108/icst.mobiquitous.2014.257921

[2] Hrvoje Benko. 2009. Beyond flat surface computing: challenges of depth-aware and curved interfaces. In *Proceedings of the 17th ACM international conference on Multimedia*. 935–944.

[3] Joanna Bergström, Tor-Salve Dalsgaard, Jason Alexander, and Kasper Hornbæk. 2021. How to evaluate object selection and manipulation in vr? guidelines from 20 years of studies. In *proceedings of the 2021 CHI conference on human factors in computing systems*. 1–20.

[4] Andrew Bragdon and Hsu-Sheng Ko. 2011. Gesture select: acquiring remote targets on large displays without pointing. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 187–196.

[5] Peter Brandl, Clifton Forlines, Daniel Wigdor, Michael Haller, and Chia Shen. 2008. Combining and measuring the benefits of bimanual pen and direct-touch interaction on horizontal interfaces. In *Proceedings of the working conference on Advanced visual interfaces*. 154–161.

[6] John Brooke. 2013. SUS: a retrospective. *Journal of usability studies* 8, 2 (2013), 29–40.

[7] Ishan Chatterjee, Robert Xiao, and Chris Harrison. 2015. Gaze+ gesture: Expressive, precise and targeted free-space interactions. In *Proceedings of the 2015 ACM on international conference on multimodal interaction*. 131–138.

[8] Ishan Chatterjee, Robert Xiao, and Chris Harrison. 2015. Gaze+Gesture: Expressive, Precise and Targeted Free-Space Interactions. In *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction* (Seattle, Washington, USA) *(ICMI '15)*. 131–138. https://doi.org/10.1145/2818346.2820752

[9] Yi Fei Cheng, Christoph Gebhardt, and Christian Holz. 2023. Interactionadapt: Interaction-driven workspace adaptation for situated virtual reality environments. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*. 1–14.

[10] Hoda Dehmeshki and Wolfgang Stuerzlinger. 2010. Design and evaluation of a perceptual-based object group selection technique. In *Proceedings of HCI 2010*. BCS Learning & Development.

[11] Unai J Fernández, Sonia Elizondo, Naroa Iriarte, Rafael Morales, Amalia Ortiz, Sebastian Marichal, Oscar Ardaiz, and Asier Marzo. 2022. A multi-object grasp technique for placement of objects in virtual reality. *Applied Sciences* 12, 9 (2022), 4193.

[12] Michael Grothaus and Erica Sadun. 2011. Interacting with Your iPad. In *Taking Your iPad to the Max, iOS 5 Edition*. Springer, 77–118.

[13] 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.

[14] John Paulin Hansen, Vijay Rajanna, I Scott MacKenzie, and Per Bækgaard. 2018. A Fitts' law study of click and dwell interaction by gaze, head and mouse with a head-mounted display. In *Proceedings of the Workshop on Communication by Gaze Interaction*. 1–5.

[15] Sandra G Hart. 2006. NASA-task load index (NASA-TLX); 20 years later. In *Proceedings of the human factors and ergonomics society annual meeting*, Vol. 50. Sage publications Sage CA: Los Angeles, CA, 904–908.

[16] Ken Hinckley, Francois Guimbretiere, Patrick Baudisch, Raman Sarin, Maneesh Agrawala, and Ed Cutrell. 2006. The springboard: multiple modes in one spring-loaded control. In *Proceedings of the SIGCHI conference on Human Factors in computing systems*. 181–190.

[17] Ken Hinckley, Randy Pausch, Dennis Proffitt, James Patten, and Neal Kassell. 1997. Cooperative bimanual action. In *Proceedings of the ACM SIGCHI Conference on Human factors in computing systems*. 27–34.

[18] Teresa Hirzle, Fabian Fischbach, Julian Karlbauer, Pascal Jansen, Jan Gugenheimer, Enrico Rukzio, and Andreas Bulling. 2022. Understanding, Addressing, and Analysing Digital Eye Strain in Virtual Reality Head-Mounted Displays. 29, 4, Article 33 (mar 2022), 80 pages. https://doi.org/10.1145/3492802

[19] Baosheng James Hou, Yasmeen Abdrabou, Florian Weidner, and Hans Gellersen. 2024. Unveiling Variations: A Comparative Study of VR Headsets Regarding Eye Tracking Volume, Gaze Accuracy, and Precision. In *2024 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)*. IEEE, 650–655.

[20] Yanfei Hu Fleischhauer, Hemant Bhaskar Surale, Florian Alt, and Ken Pfeuffer. 2023. Gaze-based mode-switching to enhance interaction with menus on tablets. In *Proceedings of the 2023 Symposium on Eye Tracking Research and Applications*. 1–8.

[21] Susu Huang, Qi Daqing, Yuan Jiabin, and Tu Huawei. 2019. Review of studies on target acquisition in virtual reality based on the crossing paradigm. *Virtual Reality & Intelligent Hardware* 1, 3 (2019), 251–264.

[22] Aulikki Hyrskykari, Howell Istance, and Stephen Vickers. 2012. Gaze gestures or dwell-based interaction?. In *Proceedings of the Symposium on Eye Tracking Research and Applications* (Santa Barbara, California) *(ETRA '12)*. 229–232. https://doi.org/10.1145/2168556.2168602

[23] Toshiya Isomoto, Shota Yamanaka, and Buntarou Shizuki. 2022. Interaction design of dwell selection toward gaze-based ar/vr interaction. In *2022 Symposium on Eye Tracking Research and Applications*. 1–2.

[24] Robert JK Jacob. 1990. What you look at is what you get: eye movement-based interaction techniques. In *Proceedings of the SIGCHI conference on Human factors in computing systems*. 11–18.

[25] Robert JK Jacob. 1991. The use of eye movements in human-computer interaction techniques: what you look at is what you get. *ACM Transactions on Information Systems (TOIS)* 9, 2 (1991), 152–169.

[26] Hyunyoung Jang, Jinwook Kim, and Jeongmi Lee. 2024. Effects of Congruent Multisensory Feedback on the Perception and Performance of Virtual Reality Hand-Retargeted Interaction. *IEEE Access* 12 (2024), 119789–119802. https://doi.org/10.1109/ACCESS.2024.3450512

[27] Jaakko Järvi and Sean Parent. 2016. One Way to Select Many. In *30th European Conference on Object-Oriented Programming (ECOOP 2016)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik.

[28] Toff Jason. 2017. Blocks: Easily Create 3D Objects in VR. https://blog.google/products/google-ar-vr/blocks-easily-create-3d-objects-vr/

[29] Jinwook Kim, Hyunyoung Jang, Dooyoung Kim, and Jeongmi Lee. 2023. Exploration of the virtual reality teleportation methods using hand-tracking, eye-tracking, and eeg. *International Journal of Human–Computer Interaction* 39, 20 (2023), 4112–4125.

[30] Mingyu Kim, Jiwon Lee, Changhun Kim, and Jinmo Kim. 2018. Tpvr: User interaction of third person virtual reality for new presence and experience. *Symmetry* 10, 4 (2018), 109.

[31] Kenrick Kin, Chengde Wan, Ken Koh, Andrei Marin, Necati Cihan Camgöz, Yubo Zhang, Yujun Cai, Fedor Kovalev, Moshe Ben-Zacharia, Shannon Hoople, et al. 2024. STMG: A Machine Learning Microgesture Recognition System for Supporting Thumb-Based VR/AR Input. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*. 1–15.

[32] Manu Kumar, Jeff Klingner, Rohan Puranik, Terry Winograd, and Andreas Paepcke. 2008. Improving the accuracy of gaze input for interaction. In *Proceedings of the 2008 symposium on Eye tracking research & applications*. 65–68.

[33] Hyeongmook Lee, Seung-Tak Noh, and Woontack Woo. 2016. Tunnelslice: Freehand subspace acquisition using an egocentric tunnel for wearable augmented reality. *IEEE Transactions on Human-Machine Systems* 47, 1 (2016), 128–139.

[34] Yang Li, Sayan Sarcar, Kibum Kim, Huawei Tu, and Xiangshi Ren. 2022. Designing successive target selection in virtual reality via penetrating the intangible interface with handheld controllers. *International Journal of Human-Computer Studies* 165 (2022), 102835.

[35] Zhipeng Li, Yi Fei Cheng, Yukang Yan, and David Lindlbauer. 2024. Predicting the Noticeability of Dynamic Virtual Elements in Virtual Reality. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*. 1–17.

[36] Daniel J. Liebling and Susan T. Dumais. 2014. Gaze and mouse coordination in everyday work. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication* (Seattle, Washington) *(UbiComp '14 Adjunct)*. 1141–1150. https://doi.org/10.1145/2638728.2641692

[37] John Finley Lucas. 2005. *Design and evaluation of 3D multiple object selection techniques*. Ph. D. Dissertation. Virginia Tech.

[38] Mathias Nørhede Lystbæk, Thorbjørn Mikkelsen, Roland Krisztandl, Eric J Gonzalez, Mar Gonzales-Franco, Hans Gellersen, and Ken Pfeuffer. 2024. Hands-on, Hands-off: Gaze-Assisted Bimanual 3D Interaction. In *UIST'24-The 37th Annual ACM Symposium on User Interface Software and Technology*. Association for Computing Machinery.

[39] Roberto A Montano-Murillo, Cuong Nguyen, Rubaiat Habib Kazi, Sriram Subramanian, Stephen DiVerdi, and Diego Martinez-Plasencia. 2020. 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)*. IEEE, 53–62.

[40] Donald A Norman. 2010. Natural user interfaces are not natural. *interactions* 17, 3 (2010), 6–10.

[41] Siyou Pei, David Kim, Alex Olwal, Yang Zhang, and Ruofei Du. 2024. UI Mobility Control in XR: Switching UI Positionings between Static, Dynamic, and Self Entities. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*. 1–12.

[42] Abdul Moiz Penkar, Christof Lutteroth, and Gerald Weber. 2012. Designing for the eye: design parameters for dwell in gaze interaction. In *Proceedings of the 24th australian computer-human interaction conference*. 479–488.

[43] Ken Pfeuffer, Yasmeen Abdrabou, Augusto Esteves, Radiah Rivu, Yomna Abdelrahman, Stefanie Meitner, Amr Saadi, and Florian Alt. 2021. ARtention: A design space for gaze-adaptive user interfaces in augmented reality. *Computers & Graphics* 95 (2021), 1–12.

[44] Ken Pfeuffer, Jason Alexander, Ming Ki Chong, and Hans Gellersen. 2014. Gaze-touch: combining gaze with multi-touch for interaction on the same surface. In *Proceedings of the 27th annual ACM symposium on User interface software and technology*. 509–518.

[45] Ken Pfeuffer, Jason Alexander, Ming Ki Chong, Yanxia Zhang, and Hans Gellersen. 2015. Gaze-Shifting: Direct-Indirect Input with Pen and Touch Modulated by Gaze. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology* (Charlotte, NC, USA) *(UIST '15)*. Association for Computing Machinery, New York, NY, USA, 373–383. https://doi.org/10.1145/2807442.2807460

[46] Ken Pfeuffer, Hans Gellersen, and Mar Gonzalez-Franco. 2024. Design Principles and Challenges for Gaze+ Pinch Interaction in XR. *IEEE Computer Graphics and Applications* 44, 3 (2024), 74–81.

[47] 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*. 99–108.

[48] Ken Pfeuffer, Jan Obernolte, Felix Dietz, Ville Mäkelä, Ludwig Sidenmark, Pavel Manakhov, Minna Pakanen, and Florian Alt. 2023. Palmgazer: Unimanual eye-hand menus in augmented reality. In *Proceedings of the 2023 ACM Symposium on Spatial User Interaction*. 1–12.

[49] Thammathip Piumsomboon, Gun Lee, Robert W Lindeman, and Mark Billinghurst. 2017. Exploring natural eye-gaze-based interaction for immersive virtual reality. In *2017 IEEE symposium on 3D user interfaces (3DUI)*. IEEE, 36–39.

[50] Alexander Plopski, Teresa Hirzle, Nahal Norouzi, Long Qian, Gerd Bruder, and Tobias Langlotz. 2022. The eye in extended reality: A survey on gaze interaction and eye tracking in head-worn extended reality. *ACM Computing Surveys (CSUR)* 55, 3 (2022), 1–39.

[51] Sara Price, Carey Jewitt, and Nikoleta Yiannoutsou. 2021. Conceptualising touch in VR. *Virtual Reality* 25, 3 (2021), 863–877.

[52] Anirudh Ramanathan. 2016. *Multi-Selection in Touch-Screen Graphical User Interfaces.* Ph. D. Dissertation.

[53] Jeff Raskin. 2000. *The humane interface: New directions for designing interactive systems.* Addison-Wesley.

[54] Jeff Sauro. 2011. *A practical guide to the system usability scale: Background, benchmarks & best practices.* Measuring Usability LLC.

[55] ShapesXR. 2024. ShapesXR — Spatial design and prototyping platform for teams. https://www.shapesxr.com/

[56] Rongkai Shi, Yushi Wei, Xuning Hu, Yu Liu, Yong Yue, Lingyun Yu, and Hai-Ning Liang. 2024. Experimental analysis of freehand multi-object selection techniques in virtual reality head-mounted displays. *Proceedings of the ACM on Human-Computer Interaction* 8, ISS (2024), 93–111.

[57] Rongkai Shi, Yushi Wei, Xueying Qin, Pan Hui, and Hai-Ning Liang. 2023. Exploring gaze-assisted and hand-based region selection in augmented reality. *Proceedings of the ACM on Human-Computer Interaction* 7, ETRA (2023), 1–19.

[58] Rongkai Shi, Jialin Zhang, Wolfgang Stuerzlinger, and Hai-Ning Liang. 2022. Group-Based Object Alignment in Virtual Reality Environments. In *Proceedings of the 2022 ACM Symposium on Spatial User Interaction.* 1–11.

[59] Rongkai Shi, Jialin Zhang, Yong Yue, Lingyun Yu, and Hai-Ning Liang. 2023. Exploration of bare-hand mid-air pointing selection techniques for dense virtual reality environments. In *Extended Abstracts of the 2023 CHI Conference on Human Factors in Computing Systems.* 1–7.

[60] Ludwig Sidenmark, Christopher Clarke, Joshua Newn, Mathias N. Lystbæk, Ken Pfeuffer, and Hans Gellersen. 2023. Vergence Matching: Inferring Attention to Objects in 3D Environments for Gaze-Assisted Selection. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems* (Hamburg, Germany) *(CHI '23)*. Association for Computing Machinery, New York, NY, USA, Article 257, 15 pages. https://doi.org/10.1145/3544548.3580685

[61] Alexandra Sipatchin, Siegfried Wahl, and Katharina Rifai. 2020. Accuracy and precision of the HTC VIVE PRO eye tracking in head-restrained and head-free conditions. *Investigative Ophthalmology & Visual Science* 61, 7 (2020), 5071–5071.

[62] Dave M Stampe and Eyal M Reingold. 1995. Selection by looking: A novel computer interface and its application to psychological research. In *Studies in visual information processing*. Vol. 6. Elsevier, 467–478.

[63] Yauhen Statsenko, Tetiana Habuza, Klaus Neidl-Van Gorkom, Nazar Zaki, and Taleb M Almansoori. 2020. Applying the inverse efficiency score to visual–motor task for studying speed-accuracy performance while aging. *Frontiers in Aging Neuroscience* 12 (2020), 574401.

[64] Anthony Steed. 2006. Towards a general model for selection in virtual environments. In *3D User Interfaces (3DUI'06)*. IEEE, 103–110.

[65] Niklas Stein, Diederick C Niehorster, Tamara Watson, Frank Steinicke, Katharina Rifai, Siegfried Wahl, and Markus Lappe. 2021. A comparison of eye tracking latencies among several commercial head-mounted displays. *i-Perception* 12, 1 (2021), 2041669520983338.

[66] Sophie Stellmach and Raimund Dachselt. 2012. Look & touch: gaze-supported target acquisition. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Austin, Texas, USA) *(CHI '12)*. Association for Computing Machinery, New York, NY, USA, 2981–2990. https://doi.org/10.1145/2207676.2208709

[67] Hemant Bhaskar Surale, Fabrice Matulic, and Daniel Vogel. 2019. Experimental analysis of barehand mid-air mode-switching techniques in virtual reality. In *Proceedings of the 2019 CHI conference on human factors in computing systems.* 1–14.

[68] JT Townsend. 1983. Stochastic modeling of elementary psychological processes. *CUP Archive* (1983).

[69] Rumeysa Turkmen, Zeynep Ecem Gelmez, Anil Ufuk Batmaz, Wolfgang Stuerzlinger, Paul Asente, Mine Sarac, Ken Pfeuffer, and Mayra Donaji Barrera Machuca. 2024. EyeGuide & EyeConGuide: Gaze-based Visual Guides to Improve 3D Sketching Systems. In *Proceedings of the CHI Conference on Human Factors in Computing Systems.* 1–14.

[70] Rumeysa Turkmen, Zeynep Ecem Gelmez, Anil Ufuk Batmaz, Wolfgang Stuerzlinger, Paul Asente, Mine Sarac, Ken Pfeuffer, and Mayra Donaji Barrera Machuca. 2024. EyeGuide & EyeConGuide: Gaze-based Visual Guides to Improve 3D Sketching Systems. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '24)*. Association for Computing Machinery, New York, NY, USA, Article 178, 14 pages. https://doi.org/10.1145/3613904.3641947

[71] Eduardo Velloso, Marcus Carter, Joshua Newn, Augusto Esteves, Christopher Clarke, and Hans Gellersen. 2017. Motion correlation: Selecting objects by matching their movement. *ACM Transactions on Computer-Human Interaction (TOCHI)* 24, 3 (2017), 1–35.

[72] Eduardo Velloso, Jayson Turner, Jason Alexander, Andreas Bulling, and Hans Gellersen. 2015. An empirical investigation of gaze selection in mid-air gestural 3D manipulation. In *Human-Computer Interaction–INTERACT 2015: 15th IFIP TC 13 International Conference, Bamberg, Germany, September 14-18, 2015, Proceedings, Part II 15.* Springer, 315–330.

[73] Uta Wagner, Mathias N Lystbæk, Pavel Manakhov, Jens Emil Sloth Grønbæk, Ken Pfeuffer, and Hans Gellersen. 2023. A fitts' law study of gaze-hand alignment for selection in 3d user interfaces. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems.* 1–15.

[74] Graham J Wills. 1996. Selection: 524,288 ways to say" this is interesting". In *Proceedings IEEE Symposium on Information Visualization'96.* IEEE, 54–60.

[75] Jacob O. Wobbrock, Meredith Ringel Morris, and Andrew D. Wilson. 2009. User-defined gestures for surface computing. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Boston, MA, USA) *(CHI '09)*. Association for Computing Machinery, New York, NY, USA, 1083–1092. https://doi.org/10.1145/1518701.1518866

[76] Dennis Wolf, Jan Gugenheimer, Marco Combosch, and Enrico Rukzio. 2020. Understanding the Heisenberg Effect of Spatial Interaction: A Selection Induced Error for Spatially Tracked Input Devices. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '20)*. 1–10. https://doi.org/10.1145/3313831.3376876

[77] Zhiqing Wu, Difeng Yu, and Jorge Goncalves. 2023. Point-and Volume-Based Multi-object Acquisition in VR. In *IFIP Conference on Human-Computer Interaction.* Springer, 20–42.

[78] Haijun Xia, Bruno Araujo, and Daniel Wigdor. 2017. Collection objects: Enabling fluid formation and manipulation of aggregate selections. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems.* 5592–5604.

[79] Difeng Yu, Qiushi Zhou, Joshua Newn, Tilman Dingler, Eduardo Velloso, and Jorge Goncalves. 2020. Fully-occluded target selection in virtual reality. *IEEE transactions on visualization and computer graphics* 26, 12 (2020), 3402–3413.

[80] Shumin Zhai, Carlos Morimoto, and Steven Ihde. 1999. Manual and gaze input cascaded (MAGIC) pointing. In *Proceedings of the SIGCHI conference on Human factors in computing systems.* 246–253.

[81] Qimeng Zhang, Chang-Hun Kim, and Hae Won Byun. 2023. Multi-finger-based arbitrary region-of-interest selection in virtual reality. *International Journal of Human–Computer Interaction* 39, 20 (2023), 3969–3983.

[82] Lixiang Zhao, Tobias Isenberg, Fuqi Xie, Hai-Ning Liang, and Lingyun Yu. 2023. MeTACAST: Target-and Context-aware Spatial Selection in VR. *IEEE Transactions on Visualization and Computer Graphics* (2023).

[83] Fengyuan Zhu, Ludwig Sidenmark, Mauricio Sousa, and Tovi Grossman. 2023. PinchLens: Applying Spatial Magnification and Adaptive Control-Display Gain for Precise Selection in Virtual Reality. In *2023 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*. IEEE, 1221–1230.