FocalPoint: Adaptive Direct Manipulation for Selecting Small 3D Virtual Objects

JIAJU MA, Brown University, United States and Stanford University, United States
JING QIAN, Brown University, United States
TONGYU ZHOU, Brown University, United States
JEFF HUANG, Brown University, United States

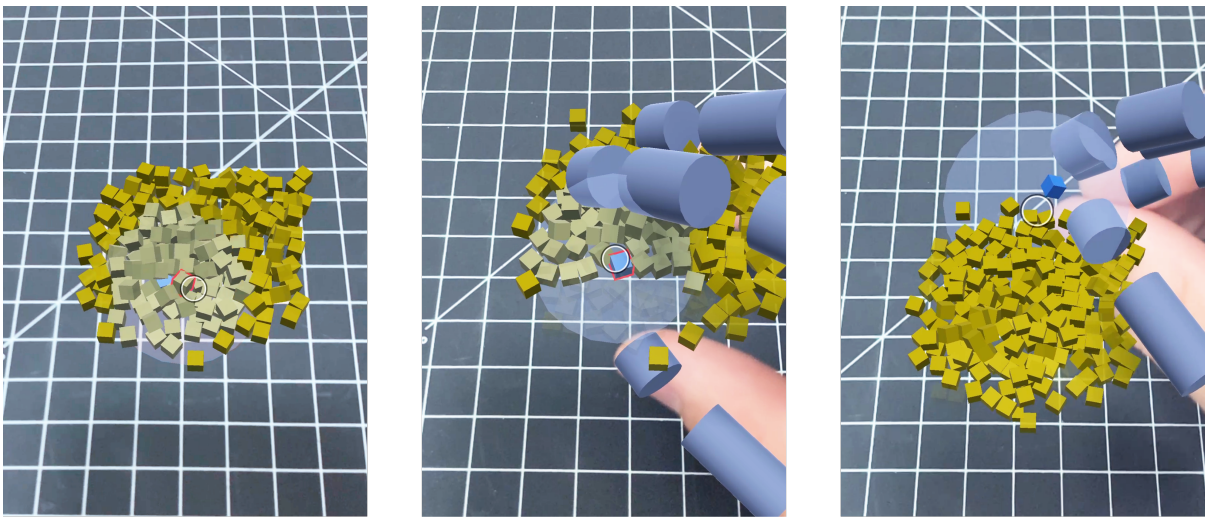

Fig. 1. FocalPoint improves the efficacy of direct manipulation selection with adaptive ray casting for small targets within reach in dense environments with many candidate objects. It continuously updates its selection cylinder based on focal regions formed by the user's selection history. An occlusion plane reveals occluded objects for direct selection. FocalPoint affords accurate bi-manual selection of occluded objects as small as 3 mm wide.

We propose FocalPoint, a direct manipulation technique in smartphone augmented reality (AR) for selecting small densely-packed objects within reach, a fundamental yet challenging task in AR due to the required accuracy and precision. FocalPoint adaptively and continuously updates a cylindrical geometry for selection disambiguation based on the user's selection history and hand movements. This design is informed by a preliminary study which revealed that participants preferred selecting objects appearing in particular regions of the screen. We evaluate FocalPoint against a baseline direct manipulation technique in a 12-participant study with two tasks: selecting a 3 mm wide target from a pile of cubes and virtually decorating a house

Authors' addresses: Jiaju Ma, jiaju_ma@alumni.brown.edu, Brown University, United States, Stanford University, United States; Jing Qian, Brown University, United States; Tongyu Zhou, Brown University, United States; Jeff Huang, Brown University, United States.

2474-9567/2023/3-ART22 $15.00
https://doi.org/10.1145/3580856

with LEGO pieces. FocalPoint was three times as accurate for selecting the correct object and 5.5 seconds faster on average; participants using FocalPoint decorated their houses more and were more satisfied with the result. We further demonstrate the finer control enabled by FocalPoint in example applications of robot repair, 3D modeling, and neural network visualizations.

CCS Concepts: • **Human-centered computing** → **Pointing**; **Gestural input**; **Mixed / augmented reality**.

Additional Key Words and Phrases: adaptive selection technique, occluded targeting, augmented reality

ACM Reference Format:
Jiaju Ma, Jing Qian, Tongyu Zhou, and Jeff Huang. 2023. FocalPoint: Adaptive Direct Manipulation for Selecting Small 3D Virtual Objects. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 7, 1, Article 22 (March 2023), 26 pages. https://doi.org/10.1145/3580856

1 INTRODUCTION

Augmented reality (AR) blends the virtual and the physical worlds by allowing *physical* inputs to interact with *virtual* objects. Free-hand direct manipulation is a popular AR interaction modality in which the user's hands engage with virtual objects to elicit a "direct" object response [47]. Although indirect 3D hand interactions like pointing and gesturing [55] may afford more precise interactions, prior work has shown that direct manipulation is fun, natural, and intuitive to users [5, 29] and has the potential to provide more nuanced interaction experiences. Therefore, **improving direct manipulation benefits how we experience AR** and unlocks more potential applications. Our proposed technique, FocalPoint, can make the most popular AR applications more immersive with fine-grained interactions beyond the screen, such as letting Pokemon Go players reach into their virtual inventory bags to grab the potion they need, or enabling Snapchat users to adjust virtual makeup or decorate their face and hair in Snap AR Lenses.

Free-hand direct manipulation is available in both headsets and smartphone AR, but the latter is more widely used[1] because it is "minimally intrusive, socially acceptable, readily available, and highly mobile [76]." Prior work [41, 70] has found that people often hold their smartphone with their non-dominant hand for on-screen interactions, leaving the other hand free for other tasks. Naturally, in the context of smartphone AR, the dominant hand becomes available to perform free-hand direct manipulation as well as other 3D hand interactions [73]. This *bimanual* interaction modality is widely employed and studied in both research literature [30, 49, 74] and in commercial products [36, 60]. By moving the interacting hand away from the smartphone screen, this bimanual approach bypasses many issues brought by the traditional 2D touch interactions, such as screen occlusion, small screen size, and the inherent limitations of 2D input for 3D interactions [73]. Moreover, the bimanual modality also enables new interaction possibilities for the interacting hand, such as holding a pen for object manipulation [65] and carrying a tiny screen for mid-air painting [51].

While recent development in direct manipulation improved both its tracking capability and interaction efficacy, efficiently selecting small densely-packed virtual objects remains a challenging task [25, 31, 64] and becomes more difficult as objects becomes smaller. While existing strategies identify virtual objects overlapping with the target selection area of the hand, there can be dozens of candidate objects; an effective technique thus needs to help the user disambiguate between them [31, 35, 45]. Furthermore, objects may *occlude* one another, especially if they are located within a pile, which pose further challenges in locating targets and specifying depth [31, 54, 61].

With FocalPoint, we focus on the efficacy of direct manipulation for selecting small virtual objects within arm's reach in smartphone AR. Our bimanual setup is based on the open source Portal-ble system [49], in which one hand moves a smartphone that overlays virtual objects on top of the physical world while the other hand interacts with them freely. A Leap Motion Controller is attached to the smartphone in this configuration, but we have also

[1]While usage is difficult to compare directly, for headsets, public sources report 520,000 HoloLens devices sold and 6,000 Magic Leap devices sold; for smartphone AR, public sources report 200,000,000 Snapchat users engage with their augmented reality features every day, and Pokemon Go has had over 1,000,000,000 installs.

developed a version of FocalPoint that requires no extra hardware besides a smartphone. It uses software-based hand tracking via the smartphone's camera (MediaPipe Hands [74]); while its performance currently does not yet reach the performance of Leap Motion's dedicated depth sensor, it is sufficient for some applications, as we demonstrate in a neural network visualizer (Section 8 and Figure 11).

To gain insights on how to design the FocalPoint technique, we conducted a preliminary study to collect empirical observations with a generic direct manipulation technique that shares commonalities with many prior works [8, 24, 49] but has documented usability issues when applied to small densely-packed objects [31, 45]. We analyze users' behaviors related to target selections and discovered that their interaction patterns tend to form **focal regions** of varying *sizes* and *locations* on the smartphone screen during selection. The findings inform an adaptive ray casting in combination with free-hand manipulation to aid object selection. This technique continuously updates the size and position of a cylindrical selection geometry to fit the user's focal region as more selections are made, allowing the selection cylinder to scope candidate objects that are more likely to be selected. We demonstrate the efficacy of FocalPoint in three example applications—mechanical parts repair, 3D modeling, and neural network visualization—expressing a range of applications that are now possible (Figure 11).

The overall work contributes a direct manipulation technique for accurately selecting small 3D objects in densely-packed piles. Our contribution is supported by the usability issues identified in the literature and behavioral findings made in the preliminary study of an existing system, and is released as an open source technique[2] for others to reproduce experiments, or extend one of its example applications.

2 RELATED WORK

2.1 Direct Manipulation for Selection

Selection is fundamental for interacting with virtual immersive environments [25, 31, 54]. A common technique is employing direct manipulation to select objects [8, 24, 49]. Defined as using gestures that act on the object itself [53], direct manipulation techniques require users to move their hands to the physical spatial location of the virtual object and perform a "grabbing" gesture for selection [26]. Because of this characteristic, this technique usually operates on virtual objects within arm's reach of the user [8, 48].

Prior work employed free-hand direct manipulation for selection tasks to enable natural engagement with virtual objects. FingARtips detects whether the user's index finger and thumb are inside a virtual object when the grabbing gesture is performed [8]. HoloDesk uses a similar physics-based approach that enables the user to pick up virtual objects with a grasping motion [24]. These techniques were designed for objects at least 50 mm in width. In general, without external physical support, users have difficulty maneuvering and holding their hands at precise locations [9, 17, 31], resulting in the selection of other objects *near* their intended objects instead [31, 35]. In addition, hand tracking limitations also make direct manipulation tasks harder than gestural command tasks because higher accuracy and granularity are required to detect the movements of individual fingers [45].

To improve the performance of selecting densely populated small objects, PRISM [18] changes the control-display ratio during hand movement to allow for finer control, but Kopper et al. noted that such non-linear mapping would cause significant mismatch between physical and perceived positions [31]. Another strategy is Balloon Selection [6], a bimanual technique in which the user holds and stretches the virtual string attached to a balloon to reach targets. In this technique, the user is not directly manipulating the selection target but the balloon proxy instead, analogous to using the mouse to control the cursor.

The improvements used in the techniques above come at the cost of changing the spatial and interaction contexts during selection, introducing additional cognitive load. We aim to provide a fluid and natural selection experience without context changing. In FocalPoint, the smartphone casts a continuously adaptive selection cylinder to narrow down candidate objects for selection and incorporates the user's interacting hand to refine

[2] https://github.com/brownhci/FocalPoint/

Table 1. A comparison of techniques that have similar features as FocalPoint. Other smartphone techniques operated on target sizes that were an order of magnitude larger [39, 63] and did not use bimanual direct manipulation. Non-smartphone techniques either used bimanual direct manipulation [6, 24] or adapted to user behavior for refining the selection [10, 16, 22], but not both. * denotes numbers that are not explicitly provided and obtained by estimating through figures.

Technique Name	Smartphone	Bimanual	Free-hand Direct Manipulation	Continuous Progressive Refinement	Behavior-driven	Occluded Object Selection	Smallest Target Size	Target Density	Action Space
SQUAD [31]						✓	100 mm	300 mm	Infinity
FingARTips [8]			✓				100 mm	50 mm	Arm Length
IntenSelect [10]					✓		60 mm	0 mm	Infinity
HoloDesk [24]		✓	✓				50 mm*	0 mm	Arm Length
Smart Ray [22]					✓	✓	7.62 mm	3.81 mm*	Infinity
IDS [45]				✓	✓		6 mm	1 mm	Arm Length
Point-and-Shake [16]						✓	5 mm	20 mm	Arm Length
Balloon Selection [6]		✓	✓				4 mm	2 mm*	Arm Length
PhoneCursor [63]	✓					✓	60 mm	200 mm	Infinity
DrillSample [39]	✓					✓	25 mm	0 mm	Infinity
FocalPoint	✓	✓	✓	✓	✓	✓	**3 mm**	**0 mm**	**Arm Length**

✓ Denotes supported attribute

and trigger selection of occluded objects as small as 3 mm wide with no gaps between objects. Table 1 compares FocalPoint to techniques that employ similar setups or mechanisms with columns and categories informed by Weise et al. [67]. While some prior systems focus on occluded object selection on smartphones [39, 63], bimanual free-hand direct manipulation [6, 24], or continuous progressive refinement [45], FocalPoint technique satisfies all proposed categories while achieving the smallest target size and one of the lowest target densities.

2.2 Ray Casting Selection Techniques

This type of technique projects a ray from the user's hand or an input device to select objects intersected with that ray [15, 33, 42]. It can become slow and inaccurate when targets are either small, located at a distance, or close to or occluded by other objects [31, 54, 61]. Specifically, selecting intended objects is difficult because small movements are amplified along the ray. When objects are densely packed, occlusion also increases the difficulty of locating and disambiguating target objects [54, 61]. To address these issues, some methods modify the ray itself to bend towards targets [10, 14] or add restrictions to where the ray can intersect [2]. Overthere [58] identifies the target position by triangulating between multiple rays. The spotlight technique [33] introduced *volumetric ray casting*, projecting a cone instead of a ray to mitigate the effects of sudden movements. Forsberg et al. [15] further developed it by adding controls to adjust the size and direction of the selection cone.

Sometimes ray casting is combined with direct manipulation to improve the user's selection experience. The Go-Go technique [48] allows the user to "cast away" the virtual hand to reach distant objects. The HOMER technique [69] uses ray casting for selection and hand interaction for manipulation. However, these works did not explore the issue of selecting densly-packed small objects, which is the focus of FocalPoint.

Inspired by existing ray casting solutions, our bimanual technique adds volumetric ray casting to enhance the direct manipulation selection experience. The shape and size of the selection cylinder is adaptively updated

based on the user's selection history. During selection, it continuously narrows down candidate objects to help improve the selection accuracy.

2.3 Behavior-Driven Selection Techniques

Other techniques employ behavioral cues to improve selection accuracy (Table 1). Hook [43] tracks how long moving objects have been followed by the user's cursor to predict potential targets. Similarly, Smart Ray [22] counts how long a ray stays near an object across time to determine the target object. SenseShapes [42] incorporates gesture and voice input, Zhang et. al [75] uses temporal patterns like taps or eye blinks, and Esteves et. al [13] employs motion matching to dynamically change selection logic based on the input modality. The Intent-Driven Selection (IDS) technique [45] attaches a selection sphere to the user's hand and uses cues like action efficiency [7, 20] and action persistence [37] to enable the selection of objects as small as 6 mm wide located 1 mm from their neighbors. However, using IDS on smaller occluded objects results in slower selection speeds as users had trouble seeing which objects were intersected by the selection sphere.

Similar in concept, we consider behavioral patterns and use the screen positions of previously selected objects to continuously and adaptively update the size and position of the selection geometry.

2.4 Selection Disambiguation

The main purpose of selection disambiguation is to determine the user's intended target from a group of possible candidates. A common mechanism used during this process is progressive refinement. First proposed by Kopper et al. [31], it divides each selection task into simpler sub-tasks until the target is selected [11]. As classified by Weise et al. [67], progressive refinement can be further divided into discrete and continuous categories. Discrete progressive refinement involves one or more distinct steps to narrow down candidate objects. For example, the SQUAD technique [31] first selects a group of objects intersected by a cone, and then the user can recursively reduce the number of objects by choosing sub-groups of objects displayed in a quad-menu. Instead of separate menus, the Expand method [9] overlays objects onto their original environments, and the Flower Ray [22] fans out objects into a flower shape.

Similarly, Disambiguation Canvas [11] and P2Roll and P2Slide [12] use a smartphone to initiate and refine selection. The Starfish method for dense environments [71] employs a 3D cursor surrounded by branches, each of which is connected to an nearby object; selection is determined when the user chooses a particular branch. Other refinement techniques do not always narrow down selection. For example, Yea Big, Yea High [25] and Slice-n-Swipe [4] also allow users to add or remove items after the initial selection. While discrete progressive refinement can be highly accurate, selection is often less fluid due to the multi-step processes and low tolerance for users' mistakes during the refining process [54]. The removal of environmental and spatial contexts in these steps further reduces immersion and makes target acquisition more difficult [45].

Continuous progressive refinement, in contrast, is fluently integrated into the selection process (Table 1). It often involves scoring, which ranks selectable objects based on scores calculated from interactions and object properties. SenseShapes [42] incorporates time, distance, visibility, and history of interactions, while IntenSelect [10] relies on locations within the cone and previous scores. Hook [43] assigns scores based on the time taken to chase after the moving target. IDS [45] also factors in the strength of intent and the movement of the user's hand.

Informed by the scoring mechanism, we use the continuous progressive refinement approach that ranks objects in situ to preserve environmental and spatial information and ensure a fluid interaction experience.

2.5 Occluded Object Selection

Various strategies have also been developed for selecting occluded objects. DrillSample [39] displays copies of objects intersected by a cast ray in different perspectives for the user to choose from. Outline Pursuit [59] matches

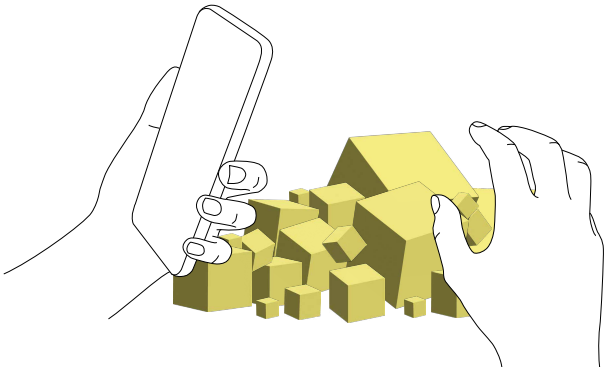

Fig. 2. An illustration of the preliminary formative study task setup. For each session, a pile of 64 virtual cubes ranging from 5 mm to 100 mm were randomly generated. Note that the number and size of the cubes are not drawn accurately.

the user's smooth pursuit eye movement to the contours of occluded targets for selection. The Control-Depth technique gradually makes objects semi-transparent to reveal previously occluded ones as the user scrolls the mouse wheel while holding down the control key [62]. In PhoneCursor [63], the user tilts the phone at different angles to control how far a vertical plane moves away from the user. Selections are then determined from objects intersecting the plane. When multiple objects are intersected by the Depth Ray [22, 64], the user can move the input device forward to select the further object, or backward the closer object. The Lock Ray [22] is similar to the Depth Ray but it allows the user to"lock" the ray in place before moving the input device. Point-and-Shake [16] implemented the Lock Ray in their setup and can select 5 mm wide spheres located 20 mm apart (see Table 1).

In our work, we attach a vertical plane to the user's interacting hand. Objects located in between the plane and the smartphone become transparent and untargetable to enable direct selection of occluded objects.

3 PRELIMINARY FORMATIVE STUDY

Continuing from the existing literature, where small object selection is relatively less tested, our initial goal is to observe natural participant behaviors during freeform selection tasks to inform our technique design. Inspired by work like DrillSample [39] and PhoneCursor [63], we use the smartphone as the source of the selection geometry cast. Because the origin, shape, and size of the selection geometry can all affect selection results [15, 33, 39], we are motivated to see if positions of virtual objects on the screen at the moment of selection is indicative of user intent and whether we can use that information to determine the selection geometry.

We recruited 10 participants (4 male and 6 female) with ages ranging from 18 to 23 ($\bar{x}$ = 20, SD = 1). 7 participants had prior experience with smartphone AR applications like Pokemon Go, but none had experience with free-hand manipulation on smartphones.

The study apparatus, created in Unity, was adopted from the open source smartphone AR system Portal-ble, which employs a generic direct manipulation method where selection is triggered by a pinch gesture. An iPhone XS Max smartphone is used with a Leap Motion Controller attached to the back for hand tracking.

During the study session, each participant was asked to select any 15 cubes from a pile of 64 cubes. The cubes, ranging from 5 mm to 100 mm wide, were randomly generated for each session within a 300 mm wide cubic space (Figure 2). Each time a cube was selected by the participants, its position was recorded in both AR world space and screen space. A point in the AR world space represents a three-dimensional point $P(x, y, z)$ in the physical world, and the smartphone screen renders that point in the screen space as a two-dimensional

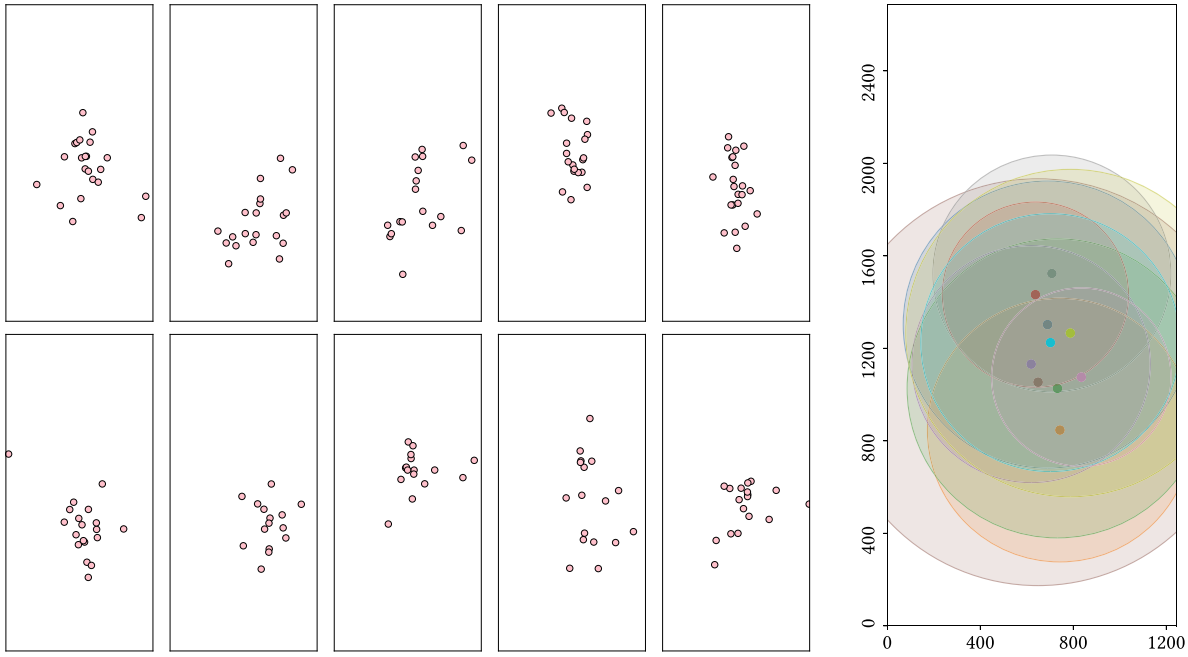

Fig. 3. *Left*: Points in each plot (1242 px × 2688 px) represents positions in screen space at which virtual objects were selected by each particular user. These points form clusters of varying sizes and indicate that the user tend to target virtual objects in a particular region of the screen for selection. *Right*: Positions of selection contained in circles. Each circle, representing a user, has a center equal to the average of all positions and a radius equal to the difference between the center and the furthest position. These regions are referred to as **focal regions** and objects placed within them are more likely to be selected.

point $P'(x, y)$. An experimenter observed and recorded participant behavior and feedback during the study. They informed the participants to think aloud and interviewed them afterwards about their overall experience.

3.1 Empirical Observations

O1: *Participants could not immediately select objects occluded by other objects in the pile and became less accurate when selecting smaller target objects.*
Some participants tried to select objects inside the pile by moving away cubes on the surface. In addition, some participants attempted to select small targets but often ended up selecting objects around it. These observations are in line with documented problems related to selecting densely-packed small objects [9, 17, 31, 35].

O2: *Participants moved the phone around in space to locate the target object before or at the same time as moving their free hand to initiate selection.*
Some participants moved the phone simultaneously with their hands, while others located an object with the phone first before reaching out their hands. Overall, participants always moved the phone (viewport) first.

O3: *Participants pointed their index finger and thumb at the target as their hand moved toward it.*
As with many direct manipulation techniques [8, 38, 49], a pinch gesture is used to confirm selections. Similar to how people reach out their hands to grab things in real life, the position and direction of their index fingers and thumbs can signal their endpoint target.

3.2 Behavior Analysis

Actual target selection occurs in 3D space but is visualized on the phone's 2D screen. The exact location of that selection on the phone's viewport was not uniformly distributed, but particular to each individual, as shown in Figure 3 (Left). In other words, the participants tend to make their selections when the targets were within a certain small region on the screen.

What is noticeable is that the selection center (the center of mass) between each individual has a modest horizontal variance, centering at $\bar{x} = 710$ px (57% from the left edge) and SD = 67 px (5% screen width). However, the vertical center varies more substantially, centering at $\bar{y} = 1,188$ px (44% from the top) and SD = 202 px (7% screen height). Centers are typically concentrated between the midpoint of the screen and the lower third of the screen. Additionally, some users are more consistent in how far from this center they place their target within the viewport, represented by the varying circle sizes. This is a crucial observation because these variances (67 px and 202 px) are *considerably large* in proportion to 3 mm wide objects, which is equivalent to 11 pixels.

Figure 3 (Right) visualizes clusters of these selections by fitting circles around them, which we will later describe how they are central to the FocalPoint technique. The center of the circles is the center of mass of where their selections occur, and the radius represents the distance between the center position and the position of the farthest point. The radii of these circles range from 385 px to 879 px, with $\bar{x} = 579$ px (46% screen width, 21% screen height) and SD = 146 px (11% screen width, 5% screen height). The standard deviation of the radii is again considerably substantial because it is almost 14 times the maximum pixel width of a 3 mm object.

In accordance with **O2**, we observe that these clusters with varying radii and centers indicate intent. When users move the phone to locate target objects, they would situate the objects in their own preferred regions on the smartphone screen before selection. Therefore, it is then reasonable to infer that objects placed within these regions would have a higher chance of being selected. We refer to such circular regions determined by the selection positions of objects as the "**focal region**."

4 DESIGN CONSIDERATIONS

Based on our preliminary study findings and inspired by prior work [10, 39, 45, 63], we summarize FocalPoint's design considerations as follows:

C1: Design a direct manipulation technique that considers the bimanual form factor with volumetric ray casting to enable accurate selections of densely-packed small objects.

C2: The focal region can indicate the user's intended objects of selection, but its size and position can vary considerably between users. To this end, our selection geometry should be continuously updated to fit the focal region, instead of being one-size-fits-all or requiring manual adjustments.

C3: FocalPoint should use a continuous progressive refinement mechanism via scoring to ensure a fluid interaction experience and preserve spatial and environmental context.

C4: FocalPoint should ease the process of locating and selecting occluded targets.

Incorporating the focal region into our selection technique allows us to address the documented flaws of pure ray casting-based or direct manipulation-based approaches (**C1**). Ray casting techniques generally operate in the two-dimensional level, meaning that objects stacked on top of each other can all be selected via a single ray cast [54, 61]. Adding the position of the user's hand into the equation gives us one more dimension to disambiguate among possible objects continuously as the selection process is being carried out (**C3**), instead of using a multi-step discrete progressive refinement mechanism [11, 54].

On the other hand, undesired selections can result from the user's inability to hold their hands at precise locations to accurately select small crowded objects [9, 17, 31]. In our bimanual form factor, the user would hold the smartphone in a steady position in order to view the virtual objects. The focal region center thus remains

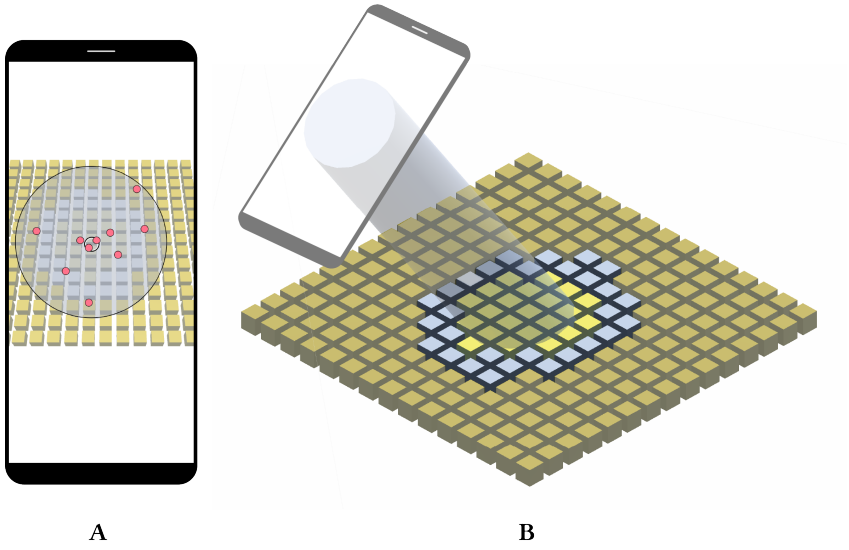

Fig. 4. *A*: The focal region is constantly updated to encapsulate the latest 10 selection positions (red circles). As more selections are made, the focal region tends to converge to a certain size and position suited to the user's preference. *B*: The selection cylinder uses the focal region as the base and extends perpendicularly outwards. The list of candidate objects includes objects within the focal region (light yellow) and located within double the focal region radius (light blue) to prevent the focal region from only shrinking.

relatively constant compared to the position of the user's interacting hand, mitigating inaccuracies caused by small and sudden movements of the hand and improving the selection experience (**C4**). The focal region is also adaptively updated to capture the nuances of individual user behavior (**C2**).

5 FOCALPOINT

As shown in Figure 1, FocalPoint is a behavior-driven direct manipulation technique that uses adaptive ray casting to improve the selection of densely-packed small objects. It uses a cylinder as the selection geometry and continuously adapts it to the user's focal region formed by their selection history. Candidate objects determined by the selection geometry are ranked according to their distances to the focal region center and the interacting hand, and the highest-ranking object is snapped to the user's hand when selection is confirmed. FocalPoint attaches an occlusion plane to the user's hand to allow direct selection of occluded objects. It also changes the opacity of objects to help users perceive depth. The following subsections are ordered to reflect the sequence of each step in the FocalPoint selection pipeline.

5.1 Selection Cylinder

The selection cylinder is used to determine a list of candidate objects that could be potentially selected and is adaptively updated to fit the user's focal region.

5.1.1 Adaptive Selection Cylinder Casting. To satisfy **C1** and **C2**, we project the focal region from the smartphone screen outwards to form a cylinder with infinite height as our selection geometry (Figure 4 B). Given that the target objects are within arm's reach, a long enough cylinder can accommodate various arm lengths. To determine the appropriate selection geometry, we experimented with various shapes such as cubes and truncated cones and

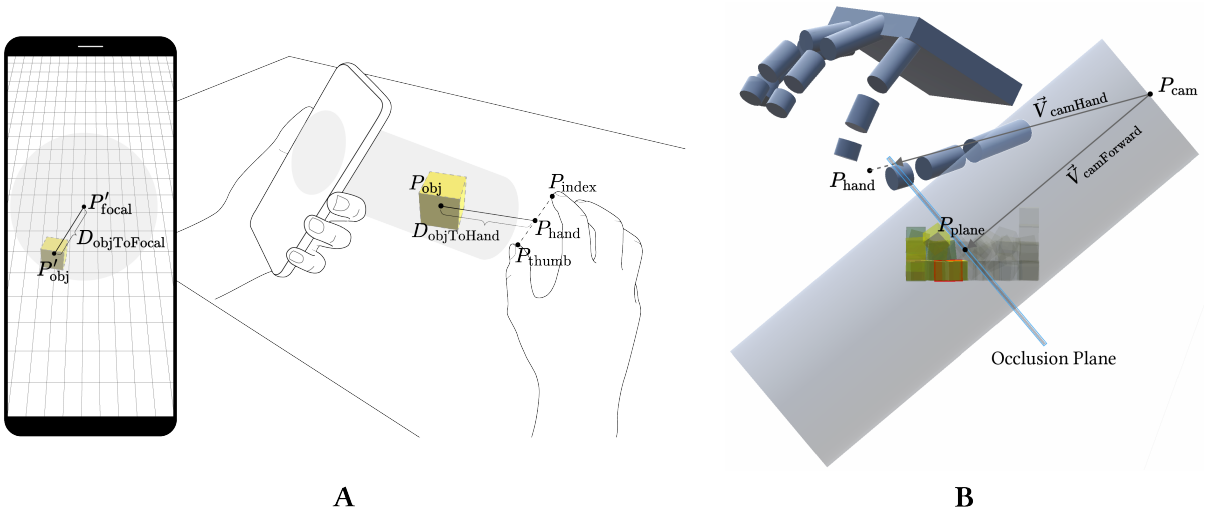

Fig. 5. *A*: Visual representation of the variables used in computing S_{priority} to determine the pre-selection object. $D_{\text{objToFocal}}$, P'_{focal}, and P'_{obj} operate in screen space while $D_{\text{objToHand}}$, P_{hand}, and P_{obj} operate in world space. *B*: The occlusion plane (light blue) is parallel to the smartphone screen and set to a small 3 mm displacement behind P_{hand}. Objects behind this plane cannot be selected and are made transparent to enable precise and direct selections of previously occluded targets.

found no noticeable difference in candidate object determination or selection disambiguation. A cylinder was used in the end because the circular focal region as the base can fully determine and adaptively update its shape without the need for a definite height.

The focal region is updated as follows. The default focal region is a circle at the center of the screen with a diameter equivalent to the diagonal length of the screen so that all objects in the viewport can be selected. After the user makes at least 10 selections, the focal region is updated to incorporate the latest 10 selection positions (Figure 4 A). The center of the focal region P'_{focal} is the average of the coordinates of these 10 points (center of mass), and the radius of the focal region is the distance between P'_{focal} and the point farthest away from P'_{focal}.

Not all prior position data are used in our algorithm because doing so includes all accidental or inaccurate selections and more recent positions can better indicate the user's current region of focus. The value 10 is determined empirically to allow the size and position of the focal region to converge relatively quickly.

5.1.2 Candidate Object Determination. Virtual objects intersecting with and on the periphery of the selection cylinder are considered **candidate objects** (Figure 4 B). Initially, selections are limited to only the objects intersecting with the cylinder, but we soon discovered that doing so resulted in the focal region becoming only smaller and smaller with each selection as newly recorded selection positions all fell within the focal region. To rectify this and allow the focal region to converge to a size most suitable for individual users, the list of candidate objects is expanded to include objects whose positions are located within double the radius of the focal region (determined empirically).

5.2 Selection Disambiguation via Scoring

Given a list of candidate objects determined by the selection cylinder, FocalPoint assigns each object a score and the highest ranking object (pre-selection object) is snapped to the user's hand when selection is confirmed.

5.2.1 Candidate Object Ranking. In accordance with **C1** and **C3**, we devise a continuous progressive refinement algorithm that assigns each candidate object a priority score S_{priority} based on its distance to the focal region center $D_{\text{objToFocal}}$ and the interacting hand $D_{\text{objToHand}}$ at each frame (Figure 5 A). The object with the highest score becomes the **pre-selection object** and is indicated to users via red contouring (see the middle panel of Figure 1). More specifically, the priority score S_{priority} is defined as

$$S_{\text{priority}} = \frac{1}{D_{\text{objToFocal}}} + \frac{1}{D_{\text{objToHand}} \cdot c_{\text{adjust}}} \tag{1}$$

As illustrated in Figure 5 A, $D_{\text{objToFocal}}$ is the distance between the position of an object P'_{obj} in screen space and the focal region center P'_{focal}, and $D_{\text{objToHand}}$ represents the distance between the position of the object P_{obj} and the user's hand P_{hand} in the world space. Since the user tend to point their index finger and thumb at a target object when making the pinch gesture for selection (**O3**), the average point between the tips of the index finger P_{index} and the thumb P_{thumb} is used as the hand position P_{hand} in our algorithm.

Since screen distance and hand distance are measured in different units, c_{adjust} is used to mitigate the difference in magnitude. In our current setup with the iPhone XS Max, $D_{\text{objToFocal}}$ ranges from 0 to 1480 (the diagonal of the phone screen width) based on the default diameter of the focal region. On the other hand, $D_{\text{objToHand}}$ takes on values in between 0 and 0.20 (0 to 20 cm from an object). This range is affected by the effective tracking range of the Leap Motion controller, which can take on different values depending on the hand tracking mechanism used. Therefore, in our current system setup, we set $c_{\text{adjust}} = 7400$. For a different smartphone with a different screen size, c_{adjust} can be adjusted proportionally.

5.2.2 Selection Confirmation. Confirming the selection of the pre-selection object requires fulfilling two criteria: the pinch gesture needs to be performed and the distance between the hand and the surface of the pre-selection object needs to be smaller than the snapping distance D_{snap}. Once both of these criteria are met, the pre-selection object is snapped to the hand position P_{hand}. We employ snapping here to remedy the inherent imprecisions of hand movements [9, 17, 31], allowing room for error to make confirming selections easier. D_{snap} is empirically determined to be 3 mm to ensure that the snapping is not noticeable to users.

5.3 Occlusion Plane

Selection tasks are more challenging with occluded target objects [31, 39, 54, 61]. We also observed this difficulty in our preliminary study (**O1**). Inspired by the vertical plane metaphor in PhoneCursor [63], an invisible **occlusion plane** parallel to the smartphone screen is employed to achieve **C4** (Figure 5 B). The plane moves with the user's hand and is offset backward from P_{hand} by D_{snap} (visualized by the short dashed line in Figure 5 B) to prevent objects intended for selection from falling behind the plane.

As the user moves their interacting hand away from the phone, objects closer to the screen fall behind the occlusion plane and are rendered transparent (alpha = 0.1), revealing the objects originally occluded by them. These transparent objects are also not selectable, and thus moving the occlusion plane also reduces the number of possible objects for selection, contributing to our selection disambiguation mechanism described above. Moreover, the occlusion plane also does not remove objects from their environmental contexts and preserves spatial relationships between the revealed objects.

5.4 Depth Indication

Depth perception is a well-known challenge for interactions in smartphone AR [5, 21]. As documented by Qian et al. [49], users often have trouble determining exactly how far they should extend their hands when virtual objects are within their reach. To address this issue, we implemented a depth indicator that changes the opacity of objects, which has been proven to be effective in indicating depth by Ping et al. [46].

In accordance with **O2**, the opacity is affected by both the smartphone position and the user's hand position, and only applies to objects intersecting with the selection cylinder. The distance between a virtual object and the smartphone $D_{\text{objToPhone}}$ primarily drives the opacity value. As the user moves the smartphone closer to an object, the object becomes more opaque. The user can bring their hand closer to that object to further increase its opacity. The opacity value only changes when $D_{\text{objToPhone}}$ is between 10 cm and 30 cm. The opacity of an object is 1.0 when $D_{\text{objToPhone}}$ is smaller than 10 cm and 0.2 when $D_{\text{objToPhone}}$ is larger than 30 cm. These thresholds are empirically determined to be the distance interval at which the user encounters trouble perceiving depth of objects within their reach.

6 EVALUATION

FocalPoint aims to improve the overall performance and user experience of direct manipulation for selecting densely populated small objects. We design a two-task study to evaluate its performance, efficacy, and user satisfaction on targets smaller than those from prior studies. Subjective feedback from participants is collected through interviews [57] to elicit both the challenges and benefits of our proposed system. We investigate the following research questions:

R1: Does FocalPoint improve the experience of direct manipulation selection for densely-packed small objects in terms of accuracy, time, and user satisfaction?

R2: What is the experience of using FocalPoint for a realistic task?

6.1 Study Design

We evaluate FocalPoint through two tasks: a controlled task where participants select a small target cube from a pile of cubes (**Task 1**), and an open-ended task that asks users to decorate a virtual house with LEGO pieces (**Task 2**). For both tasks, we measure a baseline and a FocalPoint condition per participant. The tasks are designed to evaluate FocalPoint in demanding circumstances — crowded tiny targets occluded by nearby objects. As described in Section 5, each component of FocalPoint works consecutively to process a continuous data stream. For this reason, we design the study to test FocalPoint as a holistic technique instead of separate components.

In both conditions, the same hardware setup from the preliminary study was used for its tracking reliability. Note that FocalPoint is independent of the tracking technology used as long as the user's hand is converted into an undistorted 3D space and its motions can be detected. To reduce the task learning effect, the conditions are counterbalanced between subjects for both Task 1 and Task 2. A semi-structured interview is conducted at the end of the study to elicit overall experiences, design feedback, and potential applications for FocalPoint.

6.1.1 Baseline Condition. As discussed in Related Work and shown in Table 1, existing techniques were not suitable to be directly used for baseline comparison because they either are not developed for the smartphone bimanual setup [6, 16, 22, 31], are designed to work with only large objects [8, 10, 24], or do not support free-hand direct manipulation [39, 45, 63]. Therefore, in order to evaluate how well FocalPoint improved the *free-hand direct manipulation* selection experience of small objects (R1), we incorporate existing widely-used solutions to build a baseline technique. Specifically, we add a *ranking* mechanism similar to that of IDS and Smart Ray [22, 45] to score objects by their distances to the user's interacting hand. The highest ranked object is highlighted with a 3D green contour. *Thresholding*, a common mechanism found in many direct manipulation techniques [8, 24, 49], is also used to trigger selection when the hand is pinching and the number of fingers touching the target passes a set threshold (3 in our case). For *selection disambiguation*, if multiple objects are intersected by the finger bones, the one with the most amount of bones is selected.

An alternative form of baseline is a single-hand volumetric ray casting approach, in which the selection geometry is cast from the screen but the other hand is not used in the selection process. In this technique, a

separate step is required to confirm selection, such as rotating [15, 33] or pressing on the input controller [10] from which the selection geometry is cast. An equivalent step in our smartphone setup would be rotating or tapping on the screen. In practice, this could speed up selection and result in a faster task completion time, as rotating or tapping is quicker to perform than pinching. Prior research has also shown that ray casting in general performs faster than free-hand manipulations [50, 68]. On the other hand, this additional step is likely to introduce a slight screen movement that shakes the selection geometry, negatively affecting the selection accuracy of small objects.

We did not compare FocalPoint with a volumetric ray casting technique for two reasons. First of all, as our goal is to improve the direct manipulation experience of selecting small objects on smartphone, comparing FocalPoint with a free-hand direct manipulation technique (i.e. our constructed baseline), rather than the volumetric ray casting, help us better investigate **R1**. Secondly, distance-based ranking algorithms [10, 15, 33] used in volumetric ray casting techniques consider the distances between the center ray and the virtual objects, which is similar to the ranking algorithm used in our baseline. In summary, our baseline is more suitable in helping us investigate our research questions, while incorporating certain strengths of the volumetric ray casting techniques.

6.1.2 Task 1. This task explores how participants can quickly and accurately select target objects from a collection of densely-packed, occluded small virtual objects. The IDS technique [45] could select virtual spheres 6 mm in diameter with 1 mm spaced gaps arranged in a flat 2D matrix format, and Balloon Selection [6] could select 4 mm wide virtual cubes (Table 1). However, the small targets were not occluded in either of these techniques. Informed by their study configuration, our small virtual cubes were 3 mm wide and located within a 3D pile with no gaps to demonstrate the efficacy of FocalPoint. This shape and size make our virtual object a suitable proxy for small objects commonly used in everyday life, such as screws, washes, nuts, bolts, and LEGO pieces.

In each task trial, the participant was asked to select one blue target cube from a pile of 64 otherwise yellow cubes. 18 piles were randomly generated and reused for both the baseline and FocalPoint conditions to minimize variance between them. In each pile, the position of the target cube was also randomly determined. One trial is associated with each pile, and 3 piles were used in practice trials while the remaining 15 were used in the formal trials. Within the 15 piles, 11 had the target cube partially or completely occluded by other objects. Pile orders are shuffled at the beginning of each condition. Each participant then completed 3 practice trials followed by 15 formal trials. There was a 90-second time limit per trial to reduce fatigue and avoid overburdening participants.

For each trial, we recorded time to completion and selection accuracy, defined as the number of correct selections (which is 1 in our case) divided by the number of all selections. At the end of each condition, participants also rated their satisfaction for that particular technique on a 7-point Likert scale.

6.1.3 Task 2. The second task explored how participants can use FocalPoint for real-life applications. One such activity familiar to many people is building LEGO structures. As this activity required users to accurately find and pick up specific tiny pieces from large piles, it provided an ideal scenario to test our technique.

Each participant has 5 minutes to decorate a plain house with virtual LEGO pieces of various shapes and colors, and repeats this process for each condition. A total of 61 LEGO pieces are generated within a virtual bowl next to the house, with dimensions ranging from 3 mm to 8 mm. The participant is asked to first build a pillar consisting of 11 alternating pieces next to the door (see Figure 6), after which they can freely decorate the house until time is up. An extra piece for each type of LEGO needed for the pillar is provided in case pieces were accidentally lost. Pieces placed on the house are automatically snapped to their grid positions with the correct orientation to simulate the behavior of authentic LEGO pieces.

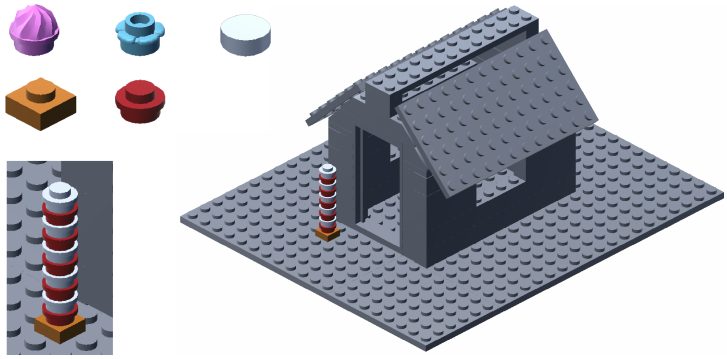

Fig. 6. *Upper Left*: Different LEGO pieces with five color variations are provided. Dimensions ranges from 3 mm to 8 mm. *Lower Left*: The 11-piece pillar that each participant is asked to construct before decorating freely. *Right*: A plain house that participants freely decorate after constructing the pillar. LEGO models are obtained through LDraw, an open standard for LEGO CAD programs [27].

6.2 Participants

We recruited 12 participants: 8 female and 4 male, 19–24 years old ($\bar{x}$ = 22, SD = 1) using convenience sampling via a student email list. 9 participants had prior experience with smartphone AR applications like Instagram filters and Pokemon Go, and 2 had limited experience with free-hand manipulation on smartphones. Participants were compensated $15 an hour for their time, with the actual average study session lasting 56 minutes.

6.3 Procedure

We conducted the study sessions in an outdoor patio to ensure safety amid the COVID-19 pandemic, following the public health guidelines established by the university, the state government, and the Centers for Disease Control and Prevention. Only one participant was present in the outdoor lab space per session and sessions were scheduled to be at least 30 minutes apart to allow ample time for cleaning. The participant was first greeted by the experimenter and informed that the session would be recorded. After reviewing and signing the consent form and safety guidelines for COVID-19, they were then ushered into our lab space. Face coverings were required for both the participant and the experimenter and social distancing was maintained at all times.

Each participant sat on a chair in front of a table around which they could freely re-position themselves. For each task, the experimenter explained the content and walked through each of the two conditions with a demo. The participant had up to 5 minutes to familiarize themselves with the task and was asked to make at least 10 selections in order to initialize the selection cylinder for FocalPoint. The participant would then complete the task trials while thinking aloud. The participant could take as many breaks as they wanted during the study. In total, we collected 384 trials (360 for Task 1, and 24 for Task 2).

7 RESULTS

7.1 Quantitative Results

7.1.1 Task 1. We log-transform the completion time and calculate accuracy scores for quantitative analysis using two-tailed t-test with data aggregated per participant. FocalPoint allows users to reach 70% accuracy on average (N = 12, $\bar{x} = 0.70, \text{SD} = 0.16$) when selecting small objects, which is almost two times more accurate ($t(11) = 10.01$, $P = 0.001$, Cohen's $d = 2.90$) than the baseline which only reaches 24% accuracy on average (N = 12, $\bar{x} = 0.24, \text{SD} = 0.11$).

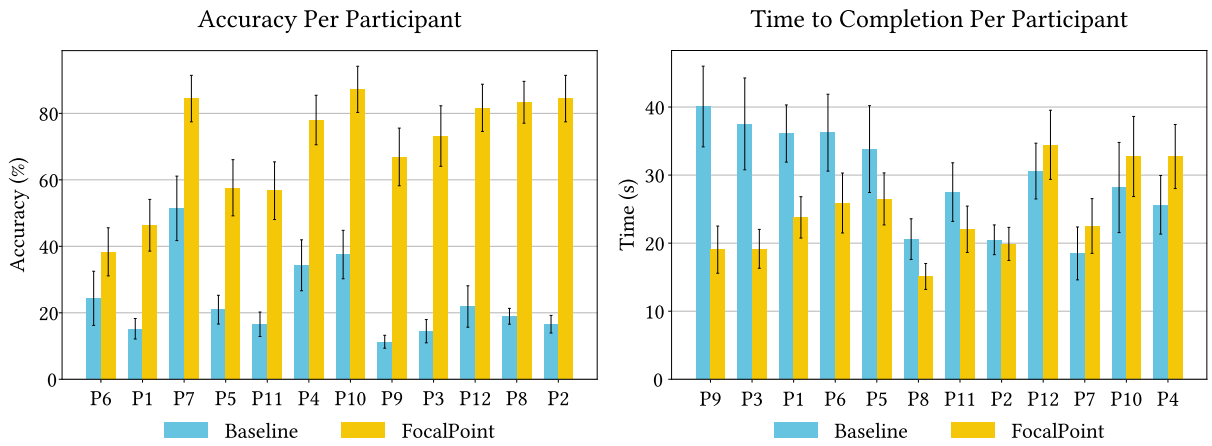

Fig. 7. *Left*: To break down the summary statistics (Section 7.1), this chart shows the accuracy and speed for each individual participant, illustrating that every participant was more accurate using FocalPoint compared to the baseline condition, often with drastic differences. *Right*: Two-thirds of the participants spent less time to complete target selections using FocalPoint in comparison to the baseline.

Overall, participants are significantly faster (5.5 seconds) with FocalPoint than using the baseline ($t(11) = -1.91$, $P = 0.04$) with a medium effect size (Cohen's $d = 0.55$). Aggregate accuracy and time distributions can be found in Figure 8 and Figure 7 shows how each participant performed using the FocalPoint compared to the baseline condition. For overall satisfaction, 83% of the participants reported that they were at least somewhat satisfied with FocalPoint, while only 17% of the participants felt the same way with the baseline conditions (Figure 8).

7.1.2 Task 2. Using FocalPoint, 9 of the 12 (75%) participants were able to complete the pillar and decorate the house. Only 6 participants (50%) were able to do the same using the baseline. Additionally, 4 (30%) participants were able to complete their decorations while they failed in their baseline conditions. All but one participant placed more decorations using FocalPoint than with the baseline condition. We present participants' finished houses to qualitatively show their task completion in Figure 9.

7.2 Qualitative Results

Six participants stated that they liked how the selection cylinder is adapted to their selection history. Specifically, P1 said "I liked how the cylinder adapts to my behavior and preferences." 4 participants also mentioned that the selection cylinder, after converging to their preferred sizes, can greatly eliminate unwanted objects from being selected and thus made them feel more confident. P10 said, "I felt more confident because I knew the selection I made wouldn't be far off." P9 thought the cylinder radius was the most optimal when it became roughly the same size as the target cube. However, other participants disagreed, and felt that the focus cylinder was too distracting and should be hidden to remove confusion. P4 specifically said that, "There was one time when I selected a few wrong objects and the cylinder expanded, and all of a sudden I felt like it becomes harder to select accurately."

7.2.1 Task 1. In the baseline condition, 6 participants mentioned that selections felt akin to "blind guessing" because it was hard to predict which objects were going to be picked up after pinching. 5 participants said it was difficult to directly locate target cubes if they were occluded by other cubes. One participant moved the phone around to find the blue cube (P6), while the other participants chose to move occluding objects away. With FocalPoint, however, 8 participants commented that it was much easier to locate target cubes by moving

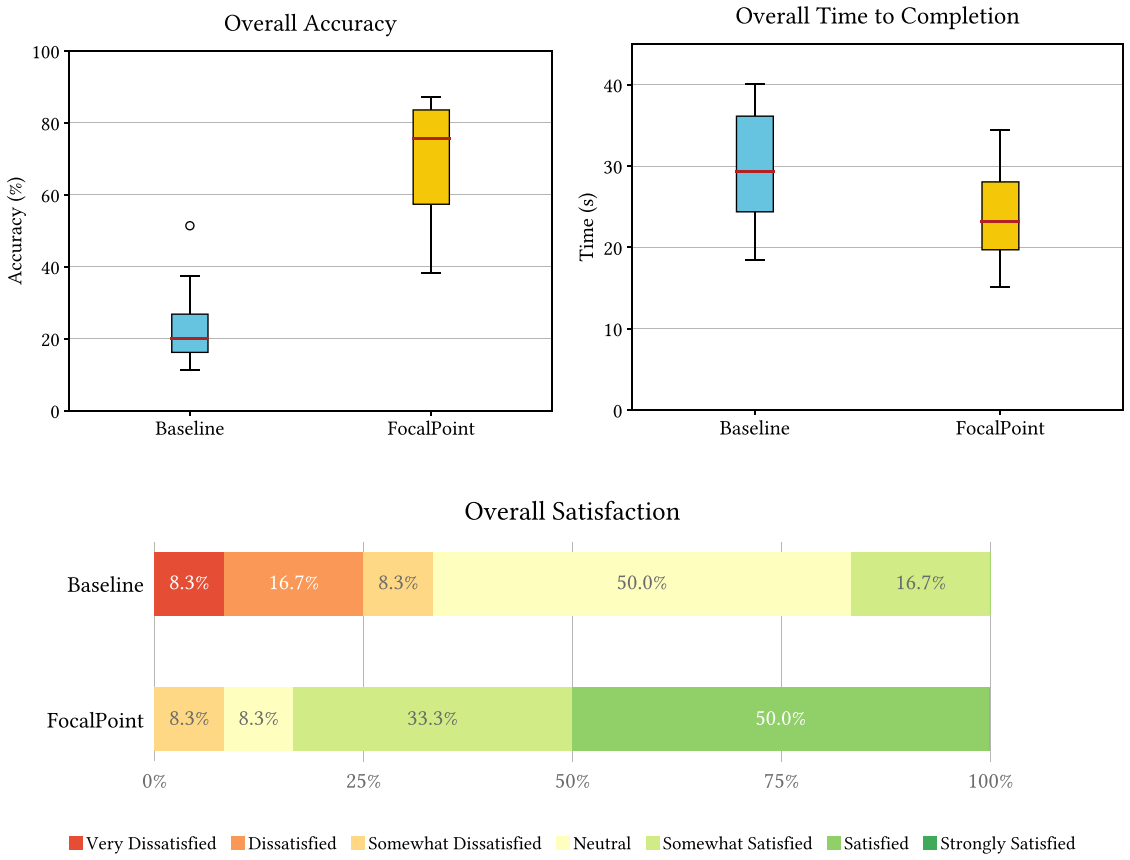

Fig. 8. Aggregate results of Task 1. *Upper Left*: Participants could select the target cube more than three times as accurately using FocalPoint. *Upper Right*: Overall, participants spent less time completing selections using FocalPoint. *Bottom*: Participants were overall more satisfied with FocalPoint in comparison to the baseline approach.

the occlusion plane. P2, P3, and P6 specifically mentioned that FocalPoint helped them "skip cubes blocking the target cube and directly select it."

7.2.2 Task 2. In general, more participants were able to complete the required pillar construction and move onto freely decorating the LEGO house with FocalPoint. 9 participants specifically mentioned that FocalPoint provided a better building experience for them than the baseline. P6 mentioned that they could use FocalPoint's occlusion plane to quickly scan the LEGO pieces in the bowl to build a "mental inventory" of where the target pieces are. P11 said that Task 2 "amplified the strength of FocalPoint" because it made repeatedly locating and selecting target objects from the bowl much easier than the baseline. P4 stated that "when I was using the baseline, after I built the pillar, I would simply grab any piece I can and place them on the roof because of the time limit. But when I was using FocalPoint, I could take time to pick and select the pieces I want to put on the house." This was also shown in P4's results, as they created a roof design with all green LEGO pieces with FocalPoint in contrast to pieces with no apparent color pattern with the baseline. However, some participants did experience frustration

Fig. 9. Decorating a virtual LEGO house is a challenge requiring participants to repeatedly locate and select different targets from a bowl filled with pieces. Overall, participants placed more decorative pieces into more intentional patterns with FocalPoint than the baseline. Rows 1 and 3 are houses decorated with FocalPoint, which are more complete compared to each house by the same participant using the baseline technique in Rows 2 and 4.

with FocalPoint; P2 observed that despite having an easier time finding target objects, "It feels somehow harder to actually select the piece I want."

7.3 Discussion

Our results demonstrated that FocalPoint performs better than the baseline for selecting densely-packed small objects in terms of accuracy, time to completion, and user satisfaction. FocalPoint also allowed participants to adorn their LEGO houses with more numerous and precisely placed decorations, illustrating its potential to be incorporated into everyday AR applications.

7.3.1 Quantitative Performance. As shown in Figure 7, all participants were more accurate with FocalPoint, and three participants even increased their accuracy by more than five-fold (P12, P8, P2). P2, P3, and P6 specifically pointed out that the occlusion plane helped with direct selection of occluded targets by allowing them to "skip cubes blocking the target cube and directly select it." In contrast, we observed that the participants often had to move away occluding objects in order to find the target in the baseline condition, contributing to the lower accuracy. Four participants mentioned that the adaptive selection cylinder was effective in ruling out non-target objects. Two participants also felt that changing cube opacity with the occlusion plane helped improve their depth perception, allowing them to "orient where my hand is spatially."

With FocalPoint, the average time to complete target selection was significantly faster (5.5 s) than that of the baseline. We believe that two factors contributed to the overall time improvement: participants can complete task trials with less selection attempts because of increased accuracy, and they spend less time finding occluded targets through the use of the occlusion plane per the experimenter's observation.

7.3.2 Selection Accuracy. Overall, FocalPoint reaches 70% accuracy on average in our evaluation. Although there is certainly room for future improvement, we believe that 70% is a reasonable accuracy when put in context with other selection techniques for selecting small or crowded targets.

First of all, FocalPoint is almost three times as accurate as the baseline technique, which is an improved direct manipulation method already incorporating widely-used techniques in the literature, including ranking, thresholding, and disambiguation mechanisms, as stated in Section 6.1.1. Secondly, even on 2D touchscreens without a z-dimension and with no hand tracking jitter, mistaken selections occur frequently when targets are densely packed. Parhi et al. have shown that accuracy in these situations is around 70% when targets are 3.8 mm wide [44]. Finally, some 3D free-hand selection techniques [23, 52] have a similar accuracy rate (in between 60% and 70%) when the target is 16 mm wide, a size much bigger than ours (3 mm wide). Similar accuracy rates can also be found in other techniques [34, 59] when objects are occluded. These results demonstrate that selecting small densely-packed targets is a very difficult task, and FocalPoint, despite not achieving the most ideal accuracy rate, makes a concrete improvement.

7.3.3 Selection Experience. To understand how FocalPoint influenced the participants' experience, we conducted a thematic analysis [28] of the qualitative feedback gathered from interviews. The participants identified various components of FocalPoint that helped them with task completions. 5 participants found that the red contour helped them foresee selection intention, as P5 commented: "the red contour makes it very obvious which piece I am going to pick up." 4 participants added that the affordance of the selection cylinder gave them greater confidence in their selection. P10 specifically mentioned that they are more confident because they knew "the selection made wouldn't be far off". The occlusion plane gave 2 participants a better sense of depth, as P7 stated that the plane "helps me with locating myself in space."

On the other hand, the participants also pointed out aspects of FocalPoint that needed to be improved. 3 participants expressed desires for the selection region to update less frequently, or only upon hand dwindle, as the constant refreshing distracted them from their overall AR experience. FocalPoint's focal region and occlusion

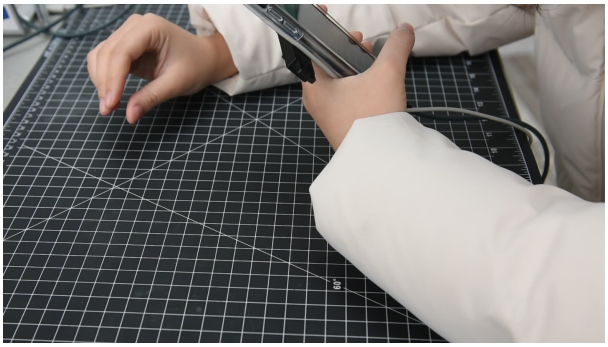

Fig. 10. Throughout the user study, the participants were found resting their arms on the tabletop when completing selection tasks, as small object selection requires precise finger and wrist movements instead of extended arm movements. As a result, the physical demand of using FocalPoint is low.

plane, while enhancing selection accuracy, might have placed additional mental burden upon users. Future systems for small, densely-packed object selection could consider subtler visualizations.

7.3.4 Physical Demand. Overall, no participant mentioned that using FocalPoint was fatiguing in Task 1, and only one participant (P2) mentioned that their right arm felt tired in Task 2. We believe that two factors contribute to this low-fatigue outcome: the nature of the selection problem and the task setup.

As shown in Figure 10, we observed that all participants in Task 1 rested their arms on the table to complete the selection tasks after briefly moving their arms around to seek an appropriate position. Because the virtual objects to be selected sat on the tabletop and were very small, the participants did not need to move and hold their arms mid-air for an extended period of time. In Task 2, more arm movements were required because the participants needed to move the LEGO pieces from the bowl to the virtual house, although we observed that the participants similarly rested their arms on the table when looking for a piece in the bowl or decorating the house with a found piece. Only P2 mentioned that their right arm (non-phone holding arm) was tired after completing Task 2, but we also noted that P2 moved and decorated the most amount of pieces (23 pieces over 5 minutes with FocalPoint, as shown in Figure 9).

Resting arms on a physical surface during interaction is a known way to reduce fatigue, as demonstrated by Balloon Selection [6] and Aperture Selection [15]. As fatigue-inducing effects like the "gorilla arm" is documented in the literature [3, 49], we want to acknowledge that FocalPoint would similarly demand higher levels of physical exertion if the tasks were not set up on a table.

7.3.5 Distant Target Selection. FocalPoint was originally designed to improve the selection resolution of objects *within reach*, as prior research [48] has shown that selecting distant objects out of the user's reach is a different class of problems. However, as distant target selection can be helpful for users interacting with the virtual environment [69], our technique can be adapted for such tasks.

Most of the FocalPoint components already work for distant objects: the selection cylinder can be trivially extended longer, and the factor $D_{\text{objToFocal}}$ in our ranking algorithm works regardless of the objects' 3D distances to the phone as it is measured in the screen space. The main modification we need is to allow the interacting hand (the hand not holding the phone) to reach distant targets, as currently the positions and motions of the virtual hand model matches exactly as those of the physical hand (Figure 5). To do so, we can adopt the non-linear mapping used in the Go-Go technique [48]. When the user extends their arm beyond 2/3 of their arm length, the virtual hand model of the interacting hand can be cast away to reach distant targets for direct manipulations.

In this way, only the mapping between positions of the physical and the virtual hand is changed, so other components like $D_{\text{objToHand}}$, the occlusion plane, and the depth cue can still function without modifications. Also, since the physical hand remains nearby the phone, the hand tracking efficacy is also not affected.

Overall, we expect the FocalPoint technique itself to perform reasonably well since the underlying mechanisms remain mostly unchanged. However, since both the target objects and the virtual hand are now at a distance, they may appear very small on the smartphone screen due to foreshortening. This can limit the distance from which this technique can be used *without changing the interaction context.* Although resolving issues like this is beyond the scope of this paper, some further explorations can be done are 1) visually enlarge the virtual objects and 2) further adjust the control-display ratio between the positions of the user's and the virtual hand to allow finer controls as demonstrated in PRISM [18].

8 EXAMPLE APPLICATIONS

Based on interviews with the participants, we implemented three example AR applications to demonstrate FocalPoint's practical feasibility: a mechanical parts repair application, a 3D modeling software, and a neural network visualizer motivated by prior work on 3D deep learning visualization [56, 72].

The first prototype application is Robotics Technician, in which the user is asked to repair a broken robot with small mechanical parts hidden inside the outer shell (Figure 11a – c). The mechanical parts are rendered to real-world scale, which become a proxy that helps the user learn how to maintain a piece of machinery. In industry, companies are applying the same idea to train technicians for real-world tasks [40]. With FocalPoint, each individual component in the robot was treated as a virtual object that could be oriented and moved around in 3D space. The user could directly examine the components inside (like an x-ray scan) and more quickly locate and select the broken parts, even when the components were occluded or in close proximity to other parts. Without FocalPoint, the user would not be able to select these interior components directly due to the obstruction of the enclosing shell. They need to either disassemble it or move it away in order to make selections inside.

The second application is a tool for creating 3D models in situ via AR. It allows the user to directly reference physical objects for scale and the environment for style (Figure 11d – f). For example, when designing a new portable camera (Figure 11f), the designer can create a sketch model that has a similar size as their hand by directly referencing it in AR without the need of manually taking measurements to get the appropriate dimensions.

Because a 3D model often consists of many small parts like knobs and handles (Figure 11f), it is difficult to precisely select them for further editing, especially when they are occluded. Moving the occluding parts away to reveal inner parts can disrupt the model structure and the user's workflow. Moreover, control points of the bounding box of a 3D model (Figure 11e), which are used for scaling and rotation, are inherently small and hard to select. Without FocalPoint, additional UI elements need to be incorporated in order for the user to work with these control points. With FocalPoint, the user can more easily select control points for scaling and rotating operations without the need of extra UI. With the occlusion plane and selection cylinder, the user can directly and accurately select occluded small parts of a model to edit further.

The third application is a tool for visualizing and modifying neural networks in AR. As deep neural networks for machine learning grow larger and deeper, prior work [56, 72] has shown that 3D visualizations can improve the transparency and interpretability of these networks. Moreover, Wang et al. has discovered that visualizing complex data in AR allows the user to walk through and around the visualizations for different views, which is fundamentally more intuitive than on-screen view manipulations [66]. However, because of the sheer size of these networks, each neuron has to be small in order for the entire network to be drawn within a reasonable realistic scale, making it difficult to select or manipulate each neuron to learn more about or modify the networks.

Our application visualizes a three-layer feedforward neural network that recognizes handwritten digits [32] with 844 cubes (Figure 11g – l). The network consists of an input layer of 784 neurons, two hidden layers (30

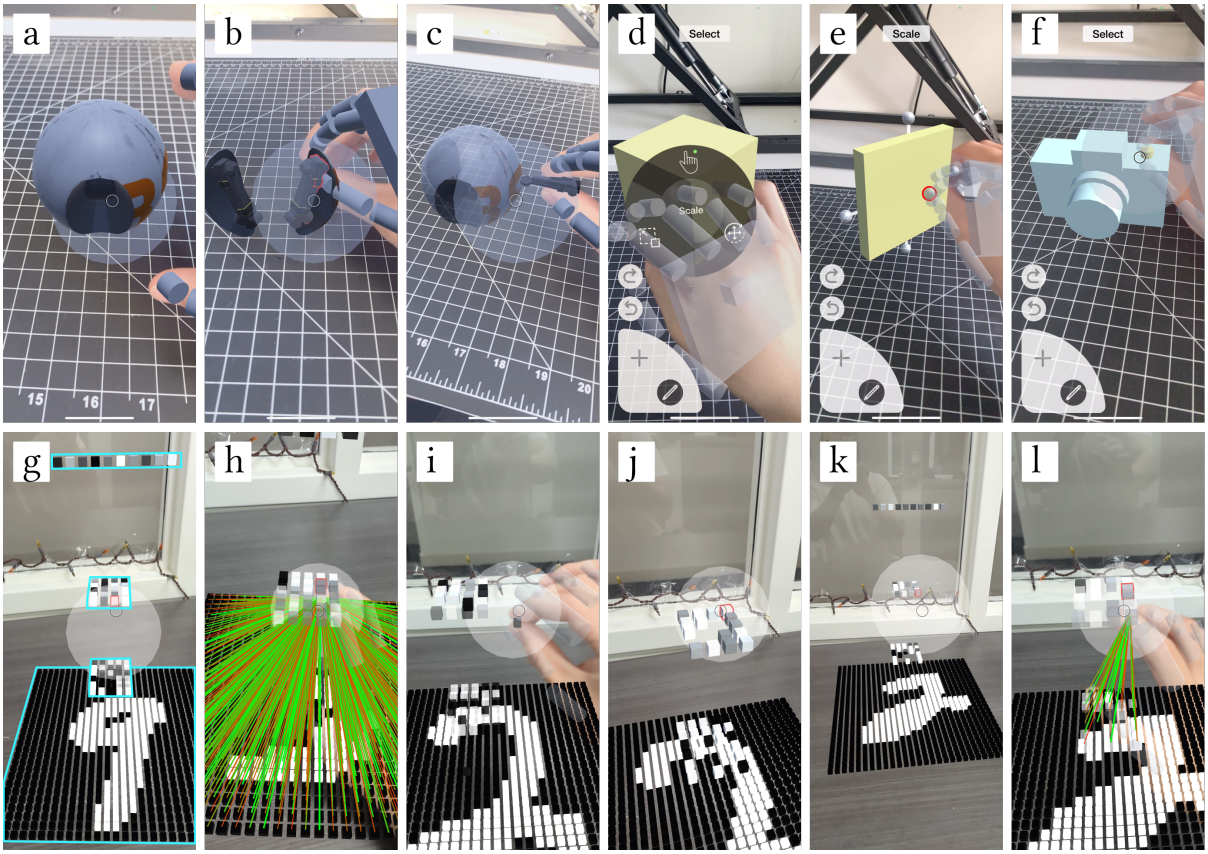

Fig. 11. *a – c*: Robotics Technician asks the user to repair a broken robot. FocalPoint allows them to directly examine the interior of the shell and quickly select problematic components without disassembling the shell. *d – f*: an AR application for creating 3D models in situ. FocalPoint enables precise selection of small control points (white dots in e) and intricate occluded parts of the model (shutter button in f) without disrupting the workflow or the model structure. *g – l*: an interactive 3D visualization of a feedforward neural network that recognizes handwritten digits. FocalPoint allows the user to select neurons to see their associated weights and biases (h and l) and remove them from layers to perform ablation studies (i – k).

neurons and 20 neurons each), and one output layer of 10 neurons (as highlighted by blue contours, from bottom to top, in Figure 11g). With FocalPoint, the user can precisely select a neuron to view their associated weights and biases to interpret the network (Figure 11h). Moreover, a common approach to evaluate a neural network is to do an ablation study in which parts of the network are removed. FocalPoint supports this by allowing the user to accurately located and select the neurons they want to remove (Figure 11i and j) and then re-evaluate the performance of the network (Figure 11k and l). This application is implemented using only an Android phone running MediaPipe Hands [74] for software-based hand tracking.

We implemented a subset of possible AR applications to demonstrate the capability of FocalPoint, but there are many other applications that could be enhanced by our technique, such as enabling users to decorate their face and hair with delicate jewelry in Snap AR Lenses, to finely adjust pixel-level details in an AR drawing app, and to precisely drop a pin onto an AR map.

9 LIMITATIONS AND FUTURE WORK

Although free-hand interaction is affected by the performance of the underlying hand tracking mechanism, the contribution of this work focuses on the interaction side. The Leap Motion setup is used to provide reliable and accurate hand tracking so that we can focus on building and evaluating our FocalPoint interaction technique.

Recent developments in machine learning-based approaches that rely only on RGB camera data [74] make it possible to perform hand tracking without external hardware [1, 19, 74], as we have implemented and demonstrated in the neural network visualizer application (Figure 11). We did not evaluate the performance of the software-only FocalPoint because it was not feasible with the latest phone hardware when the user study was conducted. Because the precision of the machine learning models improves rapidly, it is likely that phone-only hand-tracking performance will approach that of the Leap Motion version in the near term.

We focused on bimanual free-hand interaction on a smartphone and issues like fatigue arise in this form. How fatigue affects the interaction efficacy is worth investigating. Moreover, as briefly discussed in Section 6.1.1, further studies can evaluate FocalPoint against single-hand volumetric ray casting approaches to delineate performance trade-offs between direct manipulation and pointing techniques. Future work can also explore how FocalPoint and learned lessons can be adapted to other hardware like headsets. For example, the concept of a focal region may be approximated by eye tracking and the occlusion plane can be similarly attached to hands or joysticks.

10 CONCLUSION

FocalPoint focuses on changing the way selection works on small, sub-finger 3D virtual targets within reach in densely-packed piles. Its design is informed by observing user behavior in existing smartphone AR with a generic implementation in a preliminary study. Using the behavioral pattern of **focal regions**, the FocalPoint technique personalizes selection geometry based on users' selection history. It improves users' performance and experience in terms of accuracy, time, satisfaction, and qualitative outcomes for tasks such as selecting targets from a pile and placing decor on a model house. Our work provides a novel and effective interaction technique for smartphone-based free-hand direct manipulation. We apply FocalPoint to 3 applications demonstrating a range of targeting scenarios, which were not possible before with existing systems. By expanding the resolution at which users can interact with objects precisely, the most popular AR applications today can become more immersive. Augmented reality is an infinite world, but the most exciting use cases can be small.

ACKNOWLEDGMENTS

This work is supported by the National Science Foundation under Grant No. IIS-1552663.

REFERENCES

[1] Shan An, Xiajie Zhang, Dong Wei, Haogang Zhu, Jianyu Yang, and Konstantinos A. Tsintotas. 2021. Fast Monocular Hand Pose Estimation on Embedded Systems. https://doi.org/10.48550/ARXIV.2102.07067

[2] C. Andujar and F. Argelaguet. 2006. Friction surfaces: scaled ray-casting manipulation for interacting with 2D GUIs. In *Eurographics Symposium on Virtual Environments*, Ming Lin and Roger Hubbold (Eds.). The Eurographics Association. https://doi.org/10.2312/EGVE/EGVE06/101-108

[3] Myroslav Bachynskyi, Gregorio Palmas, Antti Oulasvirta, Jürgen Steimle, and Tino Weinkauf. 2015. Performance and Ergonomics of Touch Surfaces: A Comparative Study Using Biomechanical Simulation. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems* (Seoul, Republic of Korea) *(CHI '15)*. Association for Computing Machinery, New York, NY, USA, 1817–1826. https://doi.org/10.1145/2702123.2702607

[4] Felipe Bacim, Mahdi Nabiyouni, and Doug A. Bowman. 2014. Slice-n-Swipe: A free-hand gesture user interface for 3D point cloud annotation. In *2014 IEEE Symposium on 3D User Interfaces (3DUI)*. 185–186. https://doi.org/10.1109/3DUI.2014.6798882

[5] Huidong Bai, Gun A. Lee, Mukundan Ramakrishnan, and Mark Billinghurst. 2014. 3D Gesture Interaction for Handheld Augmented Reality. In *SIGGRAPH Asia 2014 Mobile Graphics and Interactive Applications* (Shenzhen, China) *(SA '14)*. Association for Computing

Machinery, New York, NY, USA, Article 7, 6 pages. https://doi.org/10.1145/2669062.2669073
[6] Hrvoje Benko and Steven Feiner. 2007. Balloon Selection: A Multi-Finger Technique for Accurate Low-Fatigue 3D Selection. In *2007 IEEE Symposium on 3D User Interfaces*. https://doi.org/10.1109/3DUI.2007.340778
[7] Elisheva Bonchek-Dokow and Gal A Kaminka. 2014. Towards computational models of intention detection and intention prediction. *Cognitive Systems Research* 28 (2014), 44–79.
[8] Volkert Buchmann, Stephen Violich, Mark Billinghurst, and Andy Cockburn. 2004. FingARtips: Gesture Based Direct Manipulation in Augmented Reality. In *Proceedings of the 2nd International Conference on Computer Graphics and Interactive Techniques in Australasia and South East Asia* (Singapore) *(GRAPHITE '04)*. Association for Computing Machinery, New York, NY, USA, 212–221. https://doi.org/10.1145/988834.988871
[9] Jeffrey Cashion, Chadwick Wingrave, and Joseph J. LaViola Jr. 2012. Dense and Dynamic 3D Selection for Game-Based Virtual Environments. *IEEE Transactions on Visualization and Computer Graphics* 18, 4 (2012), 634–642. https://doi.org/10.1109/TVCG.2012.40
[10] Gerwin de Haan, Michal Koutek, and Frits H. Post. 2005. IntenSelect: Using Dynamic Object Rating for Assisting 3D Object Selection. In *Proceedings of the 11th Eurographics Conference on Virtual Environments* (Aalborg, Denmark) *(EGVE'05)*. Eurographics Association, Goslar, DEU, 201–209.
[11] Henrique G Debarba, Jerônimo G Grandi, Anderson Maciel, Luciana Nedel, and Ronan Boulic. 2013. Disambiguation canvas: A precise selection technique for virtual environments. In *IFIP Conference on Human-Computer Interaction*. Springer, 388–405.
[12] William Delamare, Céline Coutrix, and Laurence Nigay. 2013. Mobile Pointing Task in the Physical World: Balancing Focus and Performance While Disambiguating. In *Proceedings of the 15th International Conference on Human-Computer Interaction with Mobile Devices and Services* (Munich, Germany) *(MobileHCI '13)*. Association for Computing Machinery, New York, NY, USA, 89–98. https://doi.org/10.1145/2493190.2493232
[13] Augusto Esteves, Elizabeth Bouquet, Ken Pfeuffer, and Florian Alt. 2022. One-Handed Input for Mobile Devices via Motion Matching and Orbits Controls. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 6, 2, Article 51 (jul 2022), 24 pages. https://doi.org/10.1145/3534624
[14] Alex Olwal Steven Feiner. 2003. The flexible pointer: An interaction technique for selection in augmented and virtual reality. In *Proc. UIST*, Vol. 3. 81–82.
[15] Andrew Forsberg, Kenneth Herndon, and Robert Zeleznik. 1996. Aperture Based Selection for Immersive Virtual Environments. In *Proceedings of the 9th Annual ACM Symposium on User Interface Software and Technology* (Seattle, Washington, USA) *(UIST '96)*. Association for Computing Machinery, New York, NY, USA, 95–96. https://doi.org/10.1145/237091.237105
[16] Euan Freeman, Julie Williamson, Sriram Subramanian, and Stephen Brewster. 2018. Point-and-Shake: Selecting from Levitating Object Displays. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (Montreal QC, Canada) *(CHI '18)*. Association for Computing Machinery, New York, NY, USA, 1–10. https://doi.org/10.1145/3173574.3173592
[17] Scott Frees. 2010. Context-driven interaction in immersive virtual environments. *Virtual reality* 14, 4 (2010), 277–290.
[18] Scott Frees, G. Drew Kessler, and Edwin Kay. 2007. PRISM Interaction for Enhancing Control in Immersive Virtual Environments. *ACM Trans. Comput.-Hum. Interact.* 14, 1 (may 2007), 2–es. https://doi.org/10.1145/1229855.1229857
[19] Liuhao Ge, Zhou Ren, Yuncheng Li, Zehao Xue, Yingying Wang, Jianfei Cai, and Junsong Yuan. 2019. 3D Hand Shape and Pose Estimation from a Single RGB Image. arXiv. https://doi.org/10.48550/ARXIV.1903.00812
[20] György Gergely and Gergely Csibra. 2003. Teleological reasoning in infancy: The naıve theory of rational action. *Trends in cognitive sciences* 7, 7 (2003), 287–292.
[21] Leo Gombač, Klen Čopič Pucihar, Matjaž Kljun, Paul Coulton, and Jan Grbac. 2016. 3D Virtual Tracing and Depth Perception Problem on Mobile AR. In *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems* (San Jose, California, USA) *(CHI EA '16)*. Association for Computing Machinery, New York, NY, USA, 1849–1856. https://doi.org/10.1145/2851581.2892412
[22] Tovi Grossman and Ravin Balakrishnan. 2006. The Design and Evaluation of Selection Techniques for 3D Volumetric Displays. In *Proceedings of the 19th Annual ACM Symposium on User Interface Software and Technology* (Montreux, Switzerland) *(UIST '06)*. Association for Computing Machinery, New York, NY, USA, 3–12. https://doi.org/10.1145/1166253.1166257
[23] Faizan Haque, Mathieu Nancel, and Daniel Vogel. 2015. Myopoint: Pointing and Clicking Using Forearm Mounted Electromyography and Inertial Motion Sensors. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems* (Seoul, Republic of Korea) *(CHI '15)*. Association for Computing Machinery, New York, NY, USA, 3653–3656. https://doi.org/10.1145/2702123.2702133
[24] Otmar Hilliges, David Kim, Shahram Izadi, Malte Weiss, and Andrew Wilson. 2012. HoloDesk: Direct 3d Interactions with a Situated See-through Display. 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, 2421–2430. https://doi.org/10.1145/2207676.2208405
[25] B. Jackson, B. Jelke, and G. Brown. 2018. Yea Big, Yea High: A 3D User Interface for Surface Selection by Progressive Refinement in Virtual Environments. In *2018 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*. 320–326. https://doi.org/10.1109/VR.2018.8447559
[26] Richard H. Jacoby, Mark Ferneau, and Jim Humphries. 1994. Gestural interaction in a virtual environment. In *Stereoscopic Displays and Virtual Reality Systems*, Scott S. Fisher, John O. Merritt, and Mark T. Bolas (Eds.), Vol. 2177. International Society for Optics and Photonics, SPIE, 355 – 364. https://doi.org/10.1117/12.173892
[27] James Jessiman. 1995. *LDraw, LEGO CAD software package.* LDraw Organization. https://www.ldraw.org/

[28] Michelle E Kiger and Lara Varpio. 2020. Thematic analysis of qualitative data: AMEE Guide No. 131. *Medical teacher* 42, 8 (2020), 846–854.
[29] Minseok Kim and Jae Yeol Lee. 2016. Touch and Hand Gesture-Based Interactions for Directly Manipulating 3D Virtual Objects in Mobile Augmented Reality. *Multimedia Tools Appl.* 75, 23 (dec 2016), 16529–16550. https://doi.org/10.1007/s11042-016-3355-9
[30] Shalva Kohen, Carmine Elvezio, and Steven Feiner. 2020. MiXR: A Hybrid AR Sheet Music Interface for Live Performance. In *2020 IEEE International Symposium on Mixed and Augmented Reality Adjunct (ISMAR-Adjunct)*. 76–77. https://doi.org/10.1109/ISMAR-Adjunct51615.2020.00035
[31] Regis Kopper, Felipe Bacim, and Doug A. Bowman. 2011. Rapid and accurate 3D selection by progressive refinement. In *2011 IEEE Symposium on 3D User Interfaces (3DUI)*. 67–74. https://doi.org/10.1109/3DUI.2011.5759219
[32] Yann LeCun, Corinna Cortes, and Christopher J.C. Burges. 1998. The MNIST database of handwritten digits. http://yann.lecun.com/exdb/mnist/.
[33] Jiandong Liang and Mark Green. 1994. JDCAD: A highly interactive 3D modeling system. *Computers & graphics* 18, 4 (1994), 499–506.
[34] Yiqin Lu, Chun Yu, and Yuanchun Shi. 2020. Investigating Bubble Mechanism for Ray-Casting to Improve 3D Target Acquisition in Virtual Reality. In *2020 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*. 35–43. https://doi.org/10.1109/VR46266.2020.00021
[35] John Finley Lucas. 2005. *Design and Evaluation of 3D Multiple Object Selection Techniques*. Ph.D. Dissertation. Virginia Tech.
[36] ManoMotion. 2022. *ManoMotion Gesture Technology*. ManoMotion. https://www.manomotion.com/
[37] Andrew N Meltzoff, Alison Gopnik, and Betty M Repacholi. 1999. Toddlers' understanding of intentions, desires and emotions: Explorations of the dark ages. In *Developing theories of intention: Social understanding and self-control*. Lawrence Erlbaum Associates Publishers, Mahwah, NJ, USA, 17–41.
[38] Mathias Moehring and Bernd Froehlich. 2011. Natural Interaction Metaphors for Functional Validations of Virtual Car Models. *IEEE Transactions on Visualization and Computer Graphics* 17, 9 (2011), 1195–1208. https://doi.org/10.1109/TVCG.2011.36
[39] Annette Mossel, Benjamin Venditti, and Hannes Kaufmann. 2013. DrillSample: Precise Selection in Dense Handheld Augmented Reality Environments. In *Proceedings of the Virtual Reality International Conference: Laval Virtual* (Laval, France) *(VRIC '13)*. Association for Computing Machinery, New York, NY, USA, Article 10, 10 pages. https://doi.org/10.1145/2466816.2466827
[40] Dimitris Mourtzis, Vasilios Zogopoulos, and E Vlachou. 2017. Augmented reality application to support remote maintenance as a service in the robotics industry. *Procedia Cirp* 63 (2017), 46–51.
[41] Florian Müller, Mohammadreza Khalilbeigi, Niloofar Dezfuli, Alireza Sahami Shirazi, Sebastian Günther, and Max Mühlhäuser. 2015. A Study on Proximity-Based Hand Input for One-Handed Mobile Interaction. In *Proceedings of the 3rd ACM Symposium on Spatial User Interaction* (Los Angeles, California, USA) *(SUI '15)*. Association for Computing Machinery, New York, NY, USA, 53–56. https://doi.org/10.1145/2788940.2788955
[42] A. Olwal, H. Benko, and S. Feiner. 2003. SenseShapes: using statistical geometry for object selection in a multimodal augmented reality. In *The Second IEEE and ACM International Symposium on Mixed and Augmented Reality, 2003. Proceedings.* 300–301. https://doi.org/10.1109/ISMAR.2003.1240730
[43] Michaêl Ortega. 2013. Hook: Heuristics for selecting 3D moving objects in dense target environments. In *2013 IEEE Symposium on 3D User Interfaces (3DUI)*. 119–122. https://doi.org/10.1109/3DUI.2013.6550208
[44] Pekka Parhi, Amy K. Karlson, and Benjamin B. Bederson. 2006. Target Size Study for One-Handed Thumb Use on Small Touchscreen Devices. In *Proceedings of the 8th Conference on Human-Computer Interaction with Mobile Devices and Services* (Helsinki, Finland) *(MobileHCI '06)*. Association for Computing Machinery, New York, NY, USA, 203–210. https://doi.org/10.1145/1152215.1152260
[45] Frol Periverzov and Horea Ilieş. 2015. IDS: The intent driven selection method for natural user interfaces. In *2015 IEEE Symposium on 3D User Interfaces (3DUI)*. 121–128. https://doi.org/10.1109/3DUI.2015.7131736
[46] Jiamin Ping, Bruce H. Thomas, James Baumeister, Jie Guo, Dongdong Weng, and Yue Liu. 2020. Effects of shading model and opacity on depth perception in optical see-through augmented reality. *Journal of the Society for Information Display* 28, 11 (2020), 892–904. https://doi.org/10.1002/jsid.947 arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/jsid.947
[47] Thammathip Piumsomboon, Adrian Clark, Mark Billinghurst, and Andy Cockburn. 2013. User-Defined Gestures for Augmented Reality. In *Human-Computer Interaction – INTERACT 2013*, Paula Kotzé, Gary Marsden, Gitte Lindgaard, Janet Wesson, and Marco Winckler (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 282–299.
[48] Ivan Poupyrev, Mark Billinghurst, Suzanne Weghorst, and Tadao Ichikawa. 1996. The Go-Go Interaction Technique: Non-Linear Mapping for Direct Manipulation in VR. In *Proceedings of the 9th Annual ACM Symposium on User Interface Software and Technology* (Seattle, Washington, USA) *(UIST '96)*. Association for Computing Machinery, New York, NY, USA, 79–80. https://doi.org/10.1145/237091.237102
[49] Jing Qian, Jiaju Ma, Xiangyu Li, Benjamin Attal, Haoming Lai, James Tompkin, John F. Hughes, and Jeff Huang. 2019. Portal-Ble: Intuitive Free-Hand Manipulation in Unbounded Smartphone-Based Augmented Reality. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology* (New Orleans, LA, USA) *(UIST '19)*. Association for Computing Machinery, New York, NY, USA, 133–145. https://doi.org/10.1145/3332165.3347904
[50] Jing Qian, David A. Shamma, Daniel Avrahami, and Jacob Biehl. 2020. Modality and Depth in Touchless Smartphone Augmented Reality Interactions. In *ACM International Conference on Interactive Media Experiences* (Cornella, Barcelona, Spain) *(IMX '20)*. Association for

Computing Machinery, New York, NY, USA, 74–81. https://doi.org/10.1145/3391614.3393648
[51] Jing Qian, Tongyu Zhou, Meredith Young-Ng, Jiaju Ma, Angel Cheung, Xiangyu Li, Ian Gonsher, and Jeff Huang. 2021. Portalware: Exploring Free-Hand AR Drawing with a Dual-Display Smartphone-Wearable Paradigm. In *Designing Interactive Systems Conference 2021* (Virtual Event, USA) *(DIS '21)*. Association for Computing Machinery, New York, NY, USA, 205–219. https://doi.org/10.1145/3461778.3462098
[52] Gang Ren and Eamonn O'Neill. 2013. Special Section on Touching the 3rd Dimension: 3D Selection with Freehand Gesture. *Comput. Graph.* 37, 3 (may 2013), 101–120. https://doi.org/10.1016/j.cag.2012.12.006
[53] Jaime Ruiz, Yang Li, and Edward Lank. 2011. User-Defined Motion Gestures for Mobile Interaction. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Vancouver, BC, Canada) *(CHI '11)*. Association for Computing Machinery, New York, NY, USA, 197–206. https://doi.org/10.1145/1978942.1978971
[54] Kunhee Ryu, Joong-Jae Lee, and Jung-Min Park. 2019. GG interaction: a gaze–grasp pose interaction for 3d virtual object selection. *Journal on Multimodal User Interfaces* 13, 4 (2019), 383–393.
[55] Nazmus Saquib, Rubaiat Habib Kazi, Li-Yi Wei, and Wilmot Li. 2019. Interactive Body-Driven Graphics for Augmented Video Performance. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland Uk) *(CHI '19)*. Association for Computing Machinery, New York, NY, USA, 1–12. https://doi.org/10.1145/3290605.3300852
[56] Andreas Schreiber and Marcel Bock. 2019. Visualization and Exploration of Deep Learning Networks in 3D and Virtual Reality. In *HCI International 2019 - Posters*, Constantine Stephanidis (Ed.). Springer International Publishing, Cham, 206–211.
[57] Irving Seidman. 2006. *Interviewing as qualitative research: A guide for researchers in education and the social sciences.* Teachers College Press, New York, NY.
[58] Hyunggoog Seo, Jaedong Kim, Kwanggyoon Seo, Bumki Kim, and Junyong Noh. 2021. Overthere: A Simple and Intuitive Object Registration Method for an Absolute Mid-Air Pointing Interface. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 5, 3, Article 127 (sep 2021), 24 pages. https://doi.org/10.1145/3478128
[59] Ludwig Sidenmark, Christopher Clarke, Xuesong Zhang, Jenny Phu, and Hans Gellersen. 2020. Outline Pursuits: Gaze-Assisted Selection of Occluded Objects in Virtual Reality. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '20)*. Association for Computing Machinery, New York, NY, USA, 1–13. https://doi.org/10.1145/3313831.3376438
[60] Snap Inc. 2022. *Snap AR.* Snap Inc. https://ar.snap.com/en-US
[61] Anthony Steed and Chris Parker. 2004. 3D Selection Strategies for Head Tracked and Non-Head Tracked Operation of Spatially Immersive Displays. In *8th international immersive projection technology workshop*, Vol. 2. 8 pages.
[62] Junwei Sun and Wolfgang Stuerzlinger. 2019. Selecting and Sliding Hidden Objects in 3D Desktop Environments. In *Proceedings of the 45th Graphics Interface Conference on Proceedings of Graphics Interface 2019* (Kingston, Canada) *(GI'19)*. Canadian Human-Computer Communications Society, Waterloo, CAN, Article 8, 8 pages. https://doi.org/10.20380/GI2019.08
[63] Minghui Sun, Mingming Cao, Limin Wang, and Qian Qian. 2020. PhoneCursor: Improving 3D Selection Performance With Mobile Device in AR. *IEEE Access* 8 (2020), 70616–70626. https://doi.org/10.1109/ACCESS.2020.2986037
[64] Lode Vanacken, Tovi Grossman, and Karin Coninx. 2007. Exploring the Effects of Environment Density and Target Visibility on Object Selection in 3D Virtual Environments. In *2007 IEEE Symposium on 3D User Interfaces*. https://doi.org/10.1109/3DUI.2007.340783
[65] Philipp Wacker, Oliver Nowak, Simon Voelker, and Jan Borchers. 2019. ARPen: Mid-Air Object Manipulation Techniques for a Bimanual AR System with Pen & Smartphone. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland Uk) *(CHI '19)*. Association for Computing Machinery, New York, NY, USA, 1–12. https://doi.org/10.1145/3290605.3300849
[66] Xiyao Wang, Lonni Besançon, David Rousseau, Mickael Sereno, Mehdi Ammi, and Tobias Isenberg. 2020. Towards an Understanding of Augmented Reality Extensions for Existing 3D Data Analysis Tools. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '20)*. Association for Computing Machinery, New York, NY, USA, 1–13. https://doi.org/10.1145/3313831.3376657
[67] Matthias Weise, Raphael Zender, and Ulrike Lucke. 2019. A Comprehensive Classification of 3D Selection and Manipulation Techniques. In *Proceedings of Mensch Und Computer 2019* (Hamburg, Germany) *(MuC'19)*. Association for Computing Machinery, New York, NY, USA, 321–332. https://doi.org/10.1145/3340764.3340777
[68] Matt Whitlock, Ethan Harnner, Jed R. Brubaker, Shaun Kane, and Danielle Albers Szafir. 2018. Interacting with Distant Objects in Augmented Reality. In *2018 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*. 41–48. https://doi.org/10.1109/VR.2018.8446381
[69] Curtis Wilkes and Doug A. Bowman. 2008. Advantages of Velocity-Based Scaling for Distant 3D Manipulation. In *Proceedings of the 2008 ACM Symposium on Virtual Reality Software and Technology* (Bordeaux, France) *(VRST '08)*. Association for Computing Machinery, New York, NY, USA, 23–29. https://doi.org/10.1145/1450579.1450585
[70] Pui Chung Wong, Hongbo Fu, and Kening Zhu. 2016. Back-Mirror: Back-of-Device One-Handed Interaction on Smartphones. In *SIGGRAPH ASIA 2016 Mobile Graphics and Interactive Applications* (Macau) *(SA '16)*. Association for Computing Machinery, New York, NY, USA, Article 10, 5 pages. https://doi.org/10.1145/2999508.2999522
[71] Jonathan Wonner, Jérôme Grosjean, Antonio Capobianco, and Dominique Bechmann. 2012. Starfish: A Selection Technique for Dense Virtual Environments. In *Proceedings of the 18th ACM Symposium on Virtual Reality Software and Technology* (Toronto, Ontario, Canada)

(VRST '12). Association for Computing Machinery, New York, NY, USA, 101–104. https://doi.org/10.1145/2407336.2407356

[72] Rulei Yu and Lei Shi. 2018. A user-based taxonomy for deep learning visualization. *Visual Informatics* 2, 3 (2018), 147–154. https://doi.org/10.1016/j.visinf.2018.09.001

[73] Cik Suhaimi Yusof, Huidong Bai, Mark Billinghurst, and Mohd Shahrizal Sunar. 2016. A review of 3D gesture interaction for handheld augmented reality. *Jurnal Teknologi* 78, 2-2 (2016).

[74] Fan Zhang, Valentin Bazarevsky, Andrey Vakunov, Andrei Tkachenka, George Sung, Chuo-Ling Chang, and Matthias Grundmann. 2020. MediaPipe Hands: On-device Real-time Hand Tracking. https://doi.org/10.48550/ARXIV.2006.10214

[75] Tengxiang Zhang, Xin Yi, Ruolin Wang, Jiayuan Gao, Yuntao Wang, Chun Yu, Simin Li, and Yuanchun Shi. 2019. Facilitating Temporal Synchronous Target Selection through User Behavior Modeling. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 3, 4, Article 159 (dec 2019), 24 pages. https://doi.org/10.1145/3369839

[76] Feng Zhou, Henry Been-Lirn Duh, and Mark Billinghurst. 2008. Trends in augmented reality tracking, interaction and display: A review of ten years of ISMAR. In *2008 7th IEEE/ACM International Symposium on Mixed and Augmented Reality*. 193–202. https://doi.org/10.1109/ISMAR.2008.4637362