Toward Intuitive Acquisition of Occluded VR Objects Through an Interactive Disocclusion Mini-map

Mykola Maslych* Yahya Hmaiti† Ryan Ghamandi‡ Paige Leber§ Ravi Kiran Kattoju ¶
Jacob Belga‖ Joseph J. LaViola Jr.**

ISUE Lab, University of Central Florida, 4000 Central Florida Blvd, Orlando, FL, USA, 32816

ABSTRACT

Standard selection techniques such as ray casting fail when virtual objects are partially or fully occluded. In this paper, we present two novel approaches that combine cone-casting, world-in-miniature, and grasping metaphors to disocclude objects in the representation local to the user. Through a within-subject study where we compared 4 selection techniques across 3 levels of object occlusion, we found that our techniques outperformed an alternative one that also focuses on maintaining the spatial relationships between objects. We discuss application scenarios and future research directions for these types of selection techniques.

Index Terms: Human-centered computing—Human computer interaction (HCI)—Interaction paradigms—Virtual reality; Human-centered computing—Human computer interaction (HCI)—Interaction techniques—Pointing; Human-centered computing—Human computer interaction (HCI)—Visualization—miscellaneous;

1 INTRODUCTION

Selecting objects in virtual reality should be intuitive and require little mental and physical effort [28]. Object selection methods have been studied since the early days of VR, and while significant progress has been made to simplify the selection process of clearly visible objects, occlusion imposes complications. The most widely-used selection techniques, such as ray casting and direct grabbing, fail when the objects of interest are fully or partially occluded and out of reach.

Some effort has been put into improving selections in dense and partially-occluded environments, with prominent examples being *Expand* [9] and *SQUAD* [27]. Recently Yu et al. compared multiple selection techniques and found that for rapid selection, grid-based techniques are preferable [48]. They found that participants performed best with *LassoGrid+*, where a user can outline an area and all objects within the outlined area will be organized into a grid, from which selection is done with ray casting. This works well in cases where a target is unique. However, in cases where multiple occluded objects look exactly like the target, the user has no way to tell the difference between them. In these situations, techniques that disocclude objects while preserving the spatial relationships between them are required. *GravityZone+* [48] is designed to achieve just that and was only a little slower than *LassoGrid+* in an experiment that favored more direct techniques (with a unique-looking target).

*e-mail: maslychm@gmail.com
†e-mail: yohan.hmaiti@ucf.edu
‡e-mail: ryan.ghamandi@ucf.edu
§e-mail: paige.leber@ucf.edu
¶e-mail: kattoju.ravikiran@gmail.com
‖e-mail: jacob.belga@ucf.edu
**e-mail: jjl@cs.ucf.edu

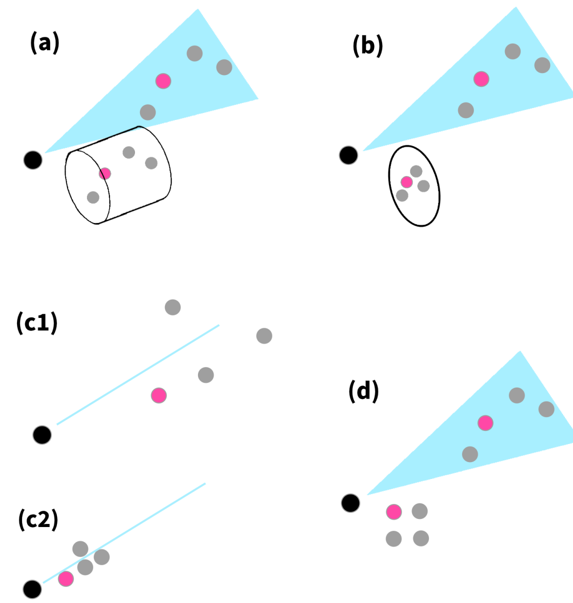

Figure 1: Selection techniques used in our user study: (a) *CylinderPIM*, (b) *DiscPIM*, (c) *GravityZone+*, (d) *ConeExpand*.

Our goal is to improve the speed and accuracy of selecting occluded targets in VR. Based on the summarized recommendations, we designed two novel selection techniques which maintain the object location information (*DiscPIM* and *CylinderPIM*). Preserving spatial relationships between the original objects helps remove ambiguity (disambiguate) when multiple objects that look the same are present in the environment. Objects which collide with a cone-shaped highlighter are displayed within a flat mini-map in the case of the *DiscPIM*, and within a cylinder mini-map in the case of the *CylinderPIM*. The abbreviation ”PIM” stands for Projective Interactive Map. The locations of the objects within the mini-maps are updated in real-time using projections of their original positions into the coordinate system of the hand that holds the highlighter. When users grab the mini-map representations of the objects, they pick up the original object from the scene.

We conducted a within-subject user study with 24 participants, where we used a version of the *Search and Repeat* task [33, 48] and we varied object density (number of selectables in a fixed-size room) and selection techniques. We implemented *ConeExpand*, a version of Expand [9] (similar to FlowerCone in [48]) to compare to our techniques. We did not implement *LassoGrid+* because it allows for a user-defined highlighted area, which went against our study design parameters where we kept the size of the selection volume fixed. We also implemented *GravityZone+* [48] as a baseline for techniques which preserve the spatial relationships between objects. We show that in a test scenario where all targets are fully-occluded, our techniques perform on par with *ConeExpand*, and outperforms *GravityZone+*. Further, we highlight practical applications of our techniques and provide implementation suggestions to practitioners.

2 Related Work

In this section, we first summarize research related to traditional target selection techniques in VR (Section 2.1). We then present prior work regarding the selection (Section 2.2) and visualization (Section 2.3) of occluded objects in VR. Lastly, we cover the literature on the design of experiments (selection tasks) and success metrics in them (Section 2.4).

2.1 Traditional Object Selection Techniques in VR

Object selection is one of the most common and frequent interactions in Virtual Environments (VEs) and many researchers have led investigations to develop fast and accurate selection techniques. Traditionally, as described by LaViola et al. [28], object selection at a distance is done with a ray that acts as a laser pointer: the first object intersecting with the ray is selected. Liang et al. suggested the use of a cone instead of a ray [29], and Forsberg et al. introduced *Aperture*, which let the user resize the selection volume with the rotation of the hand [23]. Since then we've seen multiple selection methods where collision volume suggests a list of candidate selections, and a heuristic is used to pick a single object as the intended target.

To help with selection in dynamic environments, *Intenselect* [15] calculates the distances between the cursor and each object over a sequence of frames, then uses these distances as scores for determining which object the user intended to select. *Sticky-Ray* is another selection metaphor that bends the selection ray toward the object closest to the cursor center [40]. Several techniques for target acquisition using ray and cone-based approaches were presented and summarized by Argelaguet et al. [1].

Direct manipulation is another interaction paradigm that can be applied to object selection. It relies on the virtual representation of the input device colliding with the selectable virtual objects [5]. When objects are within the distance reachable by a hand, the resizable *Bubble cursor* [25] offers reliable performance. Poupyrev et al. presented *Go-Go* [35], a technique that allows selecting objects at a distance by non-linearly scaling the distance outside of a fixed range of the hand position from the head. When the selectable objects are fully visible, all these techniques work well. However several modifications are required when the environment partially or fully occludes the targets.

2.2 Occluded Object Selection Techniques in VR

One simple way to combat occlusion is to change the point of view from which an area is observed. Mendes et al. presented *PRECIOUS* [31], which temporarily teleports the user closer to the target. Other techniques such as *BalloonProbe* [20, 21] by Elmqvist et al. tackles selection in dense and occluded VEs by temporarily moving occluding objects away from the 3D cursor. Selection by progressive refinement was also introduced as an alternative method to solve the occlusion problem, such that a group of objects is selected from the environment, then a sequence of refining steps is taken to isolate the target from the rest of the group [27]. Kopper et al. presented an acquisition technique, called *SQUAD*, that uses this concept. Sphere casting gathers a group of selectable objects into four quadrants, and pointing to one of them activates the next refining step that is repeated until only a single selectable object remains [27].

Another approach to assist with occluded object selection is through providing additional location and target characteristic awareness. Yu et al. created 3DWedge+, a visualization technique that directs the user to the locations of possible targets [47]. Montano-Murillo et al. presented a selection technique called Slicing-Volume [32], which provides a 3D volume visualization of the targets mid-air, enabling the users to inspect objects and determine target points for selection later on. Moreover, Wang et al. suggested an assistive visualization approach that maximizes the projection footprint of objects at the centre of the image while maintaining the local spatial relationships, which they named disocclusion headlight [44]. Chen et al. presented a global filtering selection technique for occluded virtual environments that incorporates a mechanic that consists of dynamic pointing. This technique is referred to as Solar-Casting, which uses a scalable semi-transparent sphere for object manipulation and then filtering for occlusion handling [11].

In an evaluation of seven VR selection techniques for fully-occluded targets, Yu et al. found that *LassoGrid+*, a technique where a participant can highlight an area of objects to be organized into a grid, results in the fastest selections and lowest error rate [48]. They found that occlusion layers, target depth, object densities along with the perception of target location impact user performance with VR object selection techniques. They also found that among the techniques which preserve the relative positions of the selectable objects, *GravityZone+* was fastest and most accurate and was only a little slower than *LassoGrid+* (approx. 4.6 seconds VS. approx. 3.7 seconds). This technique works by bringing all objects in the environment closer to and farther from the user by tilting a thumbstick on a controller, and once objects are within a certain distance away from the user, they become transparent. From there, selection is done with ray casting.

Eye tracking can also be used for object selection [17]. Wang et al. designed a selection technique that relies on eye tracking (EyeSQUAD) [45]. This technique uses selection with progressive refinement, improving selection accuracy and precision. Other techniques rely on the use of virtual hands and controllers to select targets at different levels of occlusion. For example, Schjerlund et al. suggested Ninja Hands, a selection technique for 3D virtual environments that consists of mapping the movement of a single hand to many hands, which minimizes the distance to the targets [37].

Our novel object selection techniques, *DiscPIM*, and *Cylinder-PIM* are unique because they combine elements from multiple existing selection techniques. A cone volume for highlighting an area of interest, a mini-map similar to world-in-miniature, and direct grabbing which replaces the grabbed minimized object with the original one from the virtual environment. These elements combined with a dynamic update of the mini-map give the user a real-time view of all objects within the highlighted area while still allowing for accurate selections.

2.3 Occlusion Visualization for 3D Environments

Elmqvist et al. studied visualization techniques for environments with occlusion and identified multiple disocclusion design patterns, three of which are highly relevant to our work: *Interactive Exploder*, *Virtual X-Ray*, and *Multiple Viewports* [19].

The *Interactive Exploder* is an interaction paradigm that handles occlusion through selecting an area and subsequently displacing objects in that area to be easily selectable. This results in some loss of environment context, as the spatial relationships (relative positions and rotations of objects to one another) in the exploded view are not preserved. Some of the approaches that follow this design pattern include: deformation-based volume explosion [30], 3D explosion probe [38], Expand [9], and SQUAD [27]. Some of the techniques followed the same design strategy and added modifications via applying changes to the target object by scaling, distorting, and translating [3, 6, 7, 12, 14, 18, 21, 36].

The *Virtual X-Ray* approaches use fragment shaders to handle automatically detected occlusion in the environment. Virtual objects that occlude other objects can be set to invisible or semi-transparent to allow more perspective and awareness of the occluded targets and environment. Examples of this design pattern include [2, 13, 16, 26, 42]. However, when multiple occluding surfaces and layers are present, this technique leads to misjudgments of the location of objects and their distance from the observer's view, therefore more investigation of their efficacy is necessary.

The *Multiple Viewports* design pattern displays the environment from multiple angles by adding one or more new views on the environment. This brings a more detailed perspective on the environment and is often achieved through embedding additional cameras without making any changes to the viewed images through them. Examples of this design pattern include worldlets [22], bird's eye views [24], Tumbler [36], worlds-in-miniature [41] and 3D Mini-map [47]. A notable implementation by Wang et al. [44] merges multiple viewports into a single image using graph camera [34] to maximize the selectable area of objects in the user's field of view. While accomplishing the posed task of reducing occlusion, the disadvantage of their technique is that the output image is deformed.

2.4 Design of Selection Evaluation Tasks

Since no technique is best across all environments, adequate evaluation of new, and comparison to existing selection techniques is required. Bergström et al. reviewed 20 years of VR studies with the aim of building a design framework and a checklist for future work, through evaluating the best practices across those years, suggesting recommendations based on the findings, and highlighting challenges that remain [4]. Their review focused on the evaluation of interactions in VEs through the organization of evaluation tasks and the design of the targets. Furthermore, Yu et al. proposed an evaluation framework specifically for VEs with occlusion, which works well for both distance tasks and direction tasks [47]. They also recommended using more visual cues in tasks with a large number of targets present, and using lines or highlighting of the VE orientation to improve accurate sensing of orientation and direction.

Recent work by Yu et al. compared seven selection techniques for environments and used a *Search and Repeat* task, inspired by *search+homing* that was introduced by Petford et al. [33]. This task relies on memory recall of the most recent location of the object. Each repeated trial consists of two stages: in the first stage (Search) the participant searches for an occluded target in a search area, and in the second stage (Repeat) the target is located in the same area for the participant to select the same target again. This study design is more ecologically-valid since it provides the user with information about the target location without explicitly highlighting that object, and as a result, the repeat stage task completion time represents the case when a user has some prior knowledge about the target location, even though the target is occluded.

3 Selection Techniques

Our selection techniques work with VR controllers, and they use a two-step selection process: highlight and grab. A cone-shaped highlighted volume is activated when the trigger button on the left VR controller is held down. This presents the user with a mini-map containing a real-time three-dimensional scaled-down rendering of the highlighted objects while preserving their relative positions. Subsequently, the right controller allows for the selection of the desired object by hovering over the objects directly in the mini-map. Next, the selected target object can be grabbed by pressing the secondary trigger (grab trigger) on the right controller. Finally, after the object is selected and grabbed, it disappears from the mini-map, and its original version is picked up instead. When the highlighter cone is moved across the environment, the positions and orientations of the highlighted objects are projected onto the mini-map coordinate space of the hand that holds the controller used for selection. The minimum and maximum positions local to the controller are then used to "stretch" the positions and fit the entire highlighted space in the bounds of the mini-map, using

$$\vec{P} = \frac{\vec{O}}{max(\vec{O}) - min(\vec{O})} \tag{1}$$

where $\vec{O}$ is the list of original object positions, $max()$ and $min()$ are functions that return the largest and the smallest positions from the input lists of positions (bounding box), and $\vec{P}$ is the output list of the resulting local positions of each object within the mini-map. This equation is utilized to calculate the new position of the highlighted objects during every frame for as long as the trigger button on the left controller is held down, resulting in $O(h)$ run-time, where h represents the number of highlighted objects. As a result, even if only a few objects are within the highlighted cone, and are close to the center, their projection on the mini-map will be evenly spaced-out in comparison to their actual positions in the original environment. Based on this central implementation, we developed two versions of the technique to deal with occlusion: *DiscPIM* and *CylinderPIM*.

3.1 DiscPIM

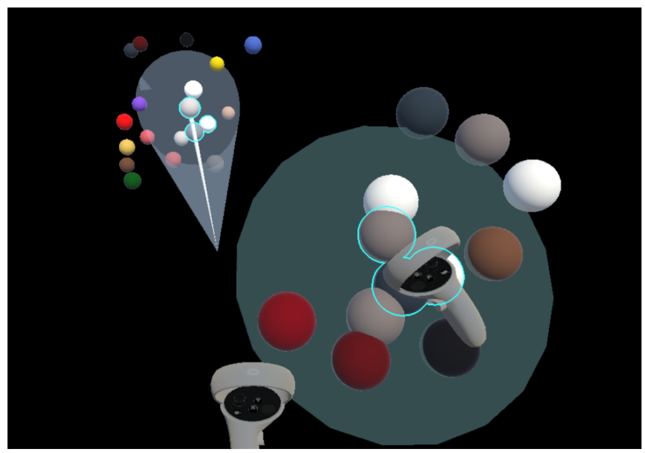

Figure 2: DiscPIM: A flat disc that follows the left controller displays dynamically updated minimized versions of the objects inside the selection cone while preserving the relative positions of selectable objects (without depth). The right controller is used to grab the objects directly from the interactive minimap. Objects that the right controller hovers over are expanded onto the circumference of the minimap, from where they can be selected directly.

DiscPIM discards the relative depths of the object copies and renders the minimized version of objects *0.07m* in diameter on a flat disc with diameter of *0.4m*. To deal with occlusion, it was designed such that when multiple objects within the mini-map overlap with the grabbing hand, the participant can opt to "freeze" the overlapping objects in slots on the circumference of the mini-map. Freezing is enabled with the trigger button on the right-hand controller while hovering over the objects in the "mini-map". The objects displayed on the circumference of the mini-map can be grabbed by using the secondary trigger button of the right controller. Figure 2 shows the *DiscPIM* as multiple objects in the environment are highlighted with a cone, and multiple objects within a minimized representation that are highlighted with a controller are occupying the circumference of the mini-map. The selection cone dimensions that we used in our user study is *1.5m* in diameter and *20m* in height.

This implementation was designed to compete with *ConeExpand*, as it also represents minimized 3D shapes of highlighted objects on a flat surface. The difference is that our mini-map preserves the X and Y relative positions between the objects, but does not automatically space out the objects into a grid-like *ConeExpand* does. *DiscPIM*, unlike *ConeExpand*, is bimanual, meaning it requires both hands to select objects.

3.2 CylinderPIM

CylinderPIM deals with occlusion by preserving the relative depth of the minimized objects *0.07m* in diameter and spacing them out within a three-dimensional cylinder *0.48m* in diameter and *0.48m* in height. The spacing-out of objects with respect to each other is the same inside the cylinder as it is in the environment. The left trigger is used to activate the highlighter cone and the hovered-over objects by the cone are displayed dynamically in the 3D cylinder. To acquire objects from inside the cylinder, the user hovers with a controller over a minimized representation of a target and presses the secondary trigger to select it.

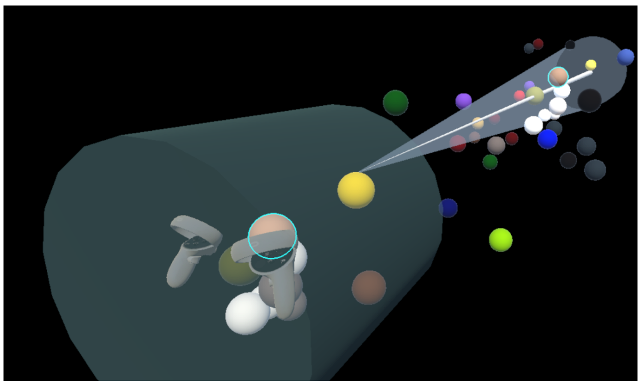

Figure 3: CylinderPIM: A cylinder that follows the left controller displays dynamically updated minimized versions of the objects inside the selection cone while preserving the relative positions of selectable objects. The right controller is used to grab the objects directly from the mini-map.

Completely preserving local spatial relationships between selectable objects allows this implementation to compete with *GravityZone+*. While the paradigms of the selections are different, our technique relied on grabbing from a mini-map, and *GravityZone+* relied on moving the original objects and selecting with a raycast. Both techniques can be used in scenarios where local positions of objects provide information crucial to selecting targets. *CylinderPIM*, unlike *GravityZone+*, requires the use of both hands to perform the selection. To keep the variations between techniques at a minimum, we used the same selection cone dimensions for this technique as for the *DiscPIM*: *1.5m* in diameter and *20m* in height.

3.3 GravityZone+

GravityZone+ [48] operates by translating all selectable objects in the environment closer or farther away from the user, such that during translation, the position and the perceived size of the objects are modified. However, the relative position of the objects is not altered in the environment. The user controls this translation by tilting the joystick on the controller, with the translation speed computed from the joystick tilt and acceleration values. Tilting the joystick forward pushes the objects away while tilting it back brings them closer. During pilot testing we set the parameters that allowed users to select the objects both quickly and comfortably. When the objects are closer than a threshold distance of *1.5m* away from the user, they become fully transparent to allow the selection of objects behind them. Selection happens with ray casting by pressing the trigger button. This technique is uni-manual; it requires one hand to operate.

The original implementation [48] states that the object speed linearly corresponded to the joystick tilt with the maximum value

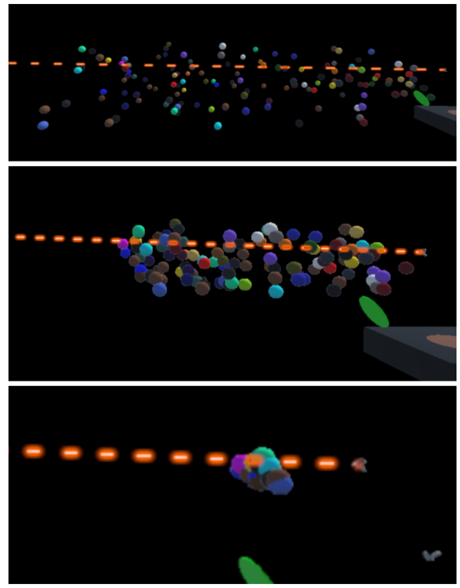

Figure 4: GravityZone+: The original positions of objects (top image) are translated towards and away from the user by tilting a joystick on a controller, with objects that are closer than a threshold distance becoming transparent. Middle image displays the translated objects after half of them became transparent due to crossing the threshold distance, bottom image displays objects translated even closer with even more of the objects being transparent. Ray casting is used for selections.

set to *6m*. During pilot testing we found that such settings made it difficult for participants to make fine adjustments, so we tuned the parameters and found a speed equation that works well

$$s' = s \times sign(x) \times x^2 \times a \tag{2}$$

where s is the base speed factor that we set to 2.5, $sign()$ is a function that returns the sign of the input, x is the raw joystick tilt on the controller's Y-axis (between -1 and 1), a is the acceleration factor, and s' is the final speed factor. The acceleration factor is computed while the joystick is held down, clamped between the values of 1 and 3, and accumulated over a second

$$a = \begin{cases} 2t+1 & \text{if } t \leq 1 \\ 3 & \text{if } t > 1 \end{cases} \tag{3}$$

where t is the time in seconds. Figure 5 shows the visual representation of the object speed equation. In basic terms, the more the joystick is tilted and the longer the time it is tilted for, the faster the objects will move. In practice, Unity runs with a variable physics framerate, so the final speed factor s' must be multiplied by the time duration of the previous frame ($Time.fixedUnscaledDeltaTime$ in Unity, denoted here as ΔT) to make the objects' speed framerate-independent. Thus outputs of Equation 2 and the following Equation

$$d = s' \times \Delta T \tag{4}$$

are used every physics update frame to compute the distance by which each object is translated in the direction of the player.

3.4 ConeExpand

ConeExpand is a two-step selection technique closely resembling Expand [9] and FlowerCone [48]. Initially, the participant uses a cone to highlight the area where the target is suspected to be. On controller trigger press, duplicates (reduced to *0.12m* in diameter) of the original objects in the highlighted area appear on a grid that

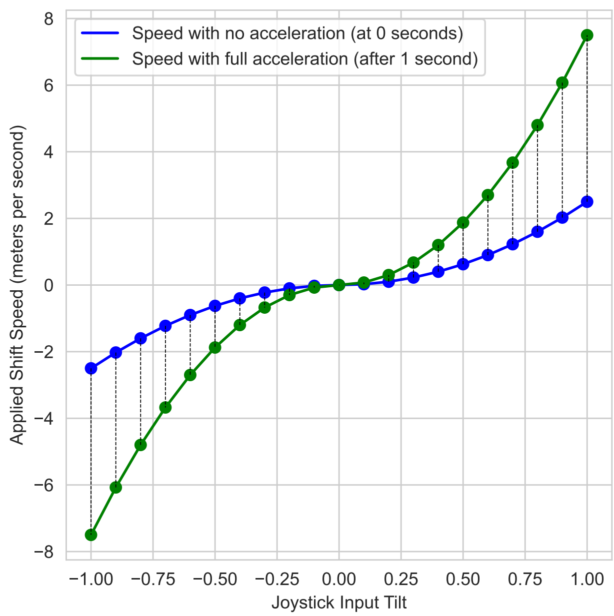

Figure 5: The object speed curve for *GravityZone+* based on the raw joystick input and the duration of current tilt. X-axis is the input, Y-axis is the output. The blue line shows the object speed at the first frame of the joystick tilt, and the green line shows the object speed after joystick has been held for 1 second and on. Black dotted lines connect the speeds that would happen at a specific timestamp at variable joystick tilt amounts.

is *1.7m* away from the user and the area selection cone is replaced with a ray. The user then can use ray casting by aiming at an object in a grid to perform selection. The grid display always faces the user, and the user has an option to discard the grid by pressing the secondary trigger button. In this case, the ray will be replaced by a cone once again and a user can start selecting an area to highlight again. Discarding the current grid is useful when a participant is scanning an area while searching for a target, or when a user does not want to perform a selection. This technique is unimanual. Just like in the other two techniques under evaluation in our user study, the selection cone diameter was set to *1.5m* and the height to *20m*, and the rest of the parameters were set based on the preferences obtained through pilot studies.

4 Experiment Framework

4.1 Hypotheses

We are interested in testing the usability of our new techniques in environments with high object density and occlusion. Our study employed a 4 by 3 within-subject design with 2 factors: Technique (*DiscPIM*, *CylinderPIM*, *ConeExpand*, *GravityZone+*) and Density (Low, Medium, High). From Low to High Density levels, the numbers of objects were 128, 256, and 512 respectively. For investigating into the selection time, we have 3 hypotheses with parts namely, (a) in the Search task and (b) in the Repeat task. Hypotheses **H1s** and **H1r** are based on the assumption that the dynamic nature of *DiscPIM*, as compared to *ConeExpand*, saves time during selection. The user simply holds the trigger button and scans the search area to locate the target. When using *GravityZone+*, which relies on a joystick to control the translation of objects, the user must wait for an object to become visible and reachable; the user might need to push back and forward the objects several times to spot the target. *CylinderPIM*, on the other hand, is dynamic and does not rely on controller gain to map joystick input to object translation, so our hypotheses **H2s** and **H2r** are that *CylinderPIM* will outperform

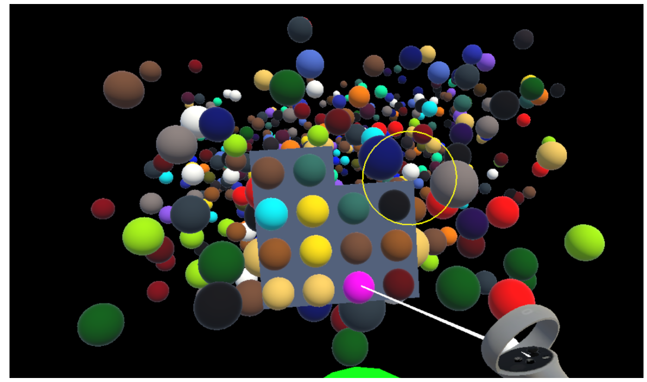

Figure 6: *ConeExpand*: In a two-step process, a user first highlights an area from which selection will be performed, and when the highlighted objects are organized into a grid the user can use ray casing to select an object from inside the grid. To discard the current grid, the user presses the secondary trigger button.

GravityZone+ in selection speed. Lastly, since we designed both of our new techniques to be dynamic and to display all highlighted objects in an accessible way, hypotheses **H3s1**, **H3s2**, **H3r1**, **H3r2** are about the Density level variable having no effect on selection speed with *DiscPIM* and *CylinderPIM*.

- **H1s**: *DiscPIM* will be faster than *ConeExpand* in Search.
- **H1r**: *DiscPIM* will be faster than *ConeExpand* in Repeat.
- **H2s**: *CylinderPIM* will be faster than *GravityZone+* in Search.
- **H2r**: *CylinderPIM* will be faster than *GravityZone+* in Repeat.
- **H3s1**: There will be no difference in user performance with *DiscPIM* in Search between different Density levels.
- **H3r1**: There will be no difference in user performance with *DiscPIM* in Repeat between different Density levels.
- **H3s2**: There will be no difference in user performance with *CylinderPIM* in Search between different Density levels.
- **H3r2**: There will be no difference in user performance with *CylinderPIM* in Repeat between different Density levels.

4.2 Environment

We used static objects in our environment which resembles environments used in experiments that tested occlusion and density [8, 48]. Participants started from a podium and directly faced the selectable objects. To combat differences in participant height, which has an effect on observed occlusion, we fixed the participant height within the environment. Dimensions of the environment were 20 meters in depth, 10 meters in width, and 5 meters in height, although participants could not judge this as the color of the walls were monotone black. A trial start button was placed in front of the podium in the same direction as the selectable objects. This button required participants to point to it with rays on both controllers and to click the trigger buttons on both controllers at the same time. During the actual selection process, *DiscPIM* and *CylinderPIM* required both hands, and *GravityZone+* and *ConeExpand* required only one hand. To have all selectable objects within their view, participants did not have to rotate their heads by more than a few degrees.

Spheres are most commonly used as the targets and distractors (selectable non-target objects) in object selection studies [4, 39, 43, 45, 48]. We ran several pilot studies, where we tested different sizes and colors of the targets and distractors, and found that a sphere radius of 0.1 meters was a good balance between complexity and applicability to real environments. Distractor selectable objects were uniformly spread out within the environment and we used easily-distinguishable colors for them. During a trial, there was a single target object uniquely colored *magenta*. We opted to differentiate

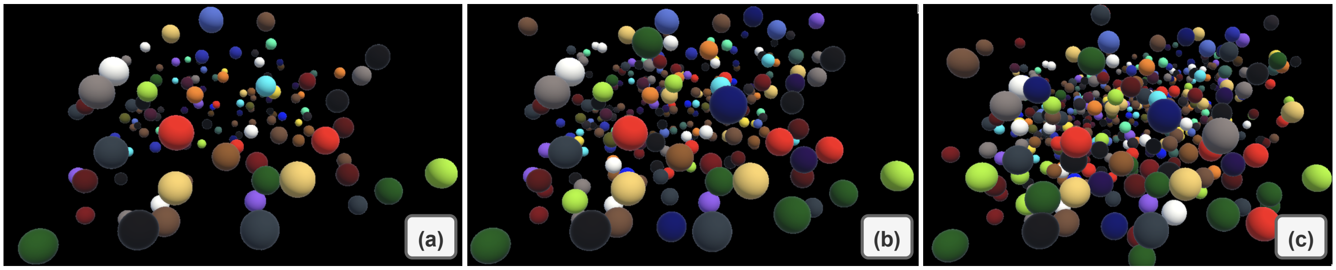

Figure 7: Participant perspective of the user study environment with three DENSITY levels: (a) Low-128 spheres, (b) Medium-256 spheres, (c) High-512 spheres.

the target from the rest of the spheres to reduce the visual search time and focus more on the selection part for our current experiment. We explored the current evaluation scenario in-depth, and we are planning a follow-up experiment where we evaluate our selection techniques compared to others from the literature in environments where distractors will be similar to the targets. In addition, locations in which the target could appear during a trial were uniformly spread out throughout the environment. We deliberately selected 30 locations for them in such a way that the target is always occluded by at least one distractor, and during an experiment, these locations were randomized without repetitions within each condition. For both *search* and *repeat* stages of each trial, a search area where a magenta target was hidden was highlighted in yellow (see Figure 8).

We implemented our user study in Unity 2020.3.29 and ran all data collection with an Oculus Quest 2 VR HMD [1] on a Windows 11 computer equipped with an Intel Core i5-11400H and RTX3060 GPU. A stable framerate of 90 frames per second was achieved during the runtime of application.

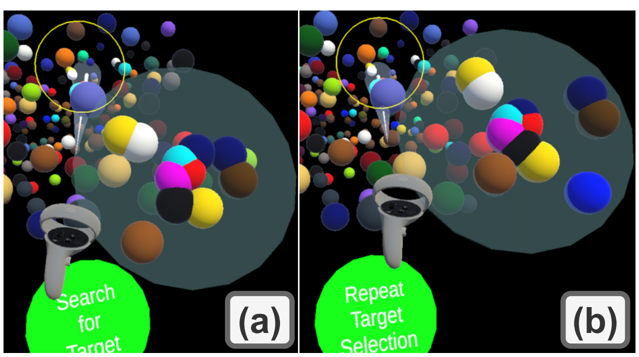

Figure 8: Participant perspective of the user study scene with the *DiscPIM* object selection technique that helps visualize the objects within a highlighted area in: (a) search task active, (b) repeat task active. The object colored in magenta in the mini-map is the target which is occluded by distractor objects.

4.3 Dependent Variables

Most commonly used metrics in object selection studies are average trial task completion time and accuracy [9, 10, 23, 48]. Task completion time is measured per trial, tracking the time from the moment a target is available, until the moment an object was selected. Trials in which the selected object is incorrect are excluded from the average completion time calculations. Accuracy per trial is usually binary - selection is considered either correct or incorrect. Alternatively, when a selected object was not correct, the trial can continue until the participant picks up the intended target object. In such a case, all trials will be successful and the most important metric becomes the target acquisition time. In our research, we are interested in this metric, so we record each trial's completion time and compute averages per condition.

4.4 Procedure

Upon arrival, prospective participants were asked to review and sign an informed consent document. After collecting participant demographics (i.e. age and gender), we introduced the participants to the Oculus Quest 2 VR HMD system they would be using throughout the study. We instructed the participants on how to adjust the headstrap and lenses to be comfortably worn and how to use the controllers to perform the required VR selection interactions. We then provided the participants with screenshots and a verbal overview of the study and what the in-VR selection tasks would involve. This included explaining that selection trials will be administered in sequences of *search and repeat* tasks, where the location of a search area and the target location during the *repeat* task will be the same as during the *search* stage that occurred right before.

After that a researcher performed the technique and task completion live and the participant was able to view the researcher's point of view of the environment and the technique on an external monitor. The researcher provided a verbal explanation while performing the technique live and the participants were allowed to ask questions about using the technique. Once all their questions were answered, the participants were allowed to practice the search and repeat tasks in the medium density environment (256 spheres) until they felt comfortable with the technique (usually around 1 and a half minutes). At that time the participants told the researcher that they have had enough practice and are ready to start the data collection. During pilot studies we found that the live demonstration, answering participants' questions, and a practice session for each selection technique were sufficient to give each technique a fair chance at performing well.

After the familiarization stage, participants used a technique in three density levels administered in random order. For each density level, five trials of the search and repeat tasks were performed with a unique occluded target location for each search and repeat task pair. During these trials, participants first searched for an occluded target of magenta color within an area marked with a yellow circle (search task), and after successfully selecting the found target, they repeated the same selection, this time with the awareness of the target location in the environment. After successfully completing both the search and repeat tasks, a new target location and search area were set for the next trial. After the in-VR tasks for each selection technique, participants were administered a SUS and NASA-TLX surveys. After a short break, participants were required to complete the same steps for the remaining selection techniques, a total of 4 times (the order of techniques was counterbalanced in order to avoid learning effects). The study took approximately 50 minutes, and

[1] https://www.meta.com/quest/products/quest-2/

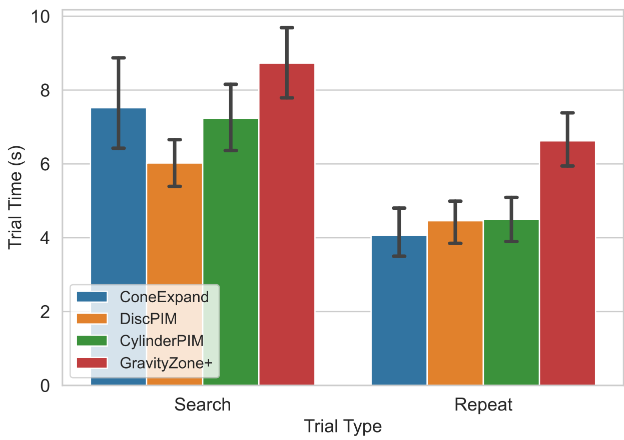

Figure 9: Average trial time in seconds for each TECHNIQUE across search and repeat trial types. Confidence intervals are 95%.

participants were compensated $10 USD in cash for their time.

4.5 Participants

For our study, we recruited participants from the University of Central Florida. Participants were required to be 18 years of age or older, have normal or corrected-to-normal vision and normal hearing, with ability to walk, extend both arms, use both hands, and speak and understand English. Participants were also required to have no known visual, auditory, neuropathic, or physical disabilities. Our final participant pool consisted of 24 people (16 male, 7 female, 1 genderfluid) of ages ranging from 18 to 35, with a mean age of 23.6 years old.

5 RESULTS

To analyze the task completion time, we first discarded outliers (35 trials, 1.22%), the trials where selection time was above four standard deviations from the mean ($> \mu + 4\sigma$) per condition. Such results occur due to participants' distraction during the experiment and they might skew results. We decided not to discard trials that took more than one attempt to complete. The selection time data was not normally distributed (based on Shapiro-Wilks test), so it underwent processing with ART [46] before two-way RM-ANOVA was applied. Search and repeat trial times were analyzed separately. For pairwise comparisons we applied ART-C [46], and ran Bonferroni-adjusted pairwise comparisons using t-tests.

Figure 9 displays the overall results for search and repeat tasks across all levels of DENSITY averaged. In search, mean trial time is 6.02 seconds for *DiscPIM*, 7.23 seconds for *CylinderPIM*, 7.52 seconds for *ConeExpand*, and 8.73 seconds for *GravityZone+*. The effect of TECHNIQUE on search completion time was significant ($F_{2.206,50.727} = 9.714$, $p < .001$), and the effect of DENSITY on search completion time was also significant ($F_{2,46} = 17.454$, $p < .001$). For search trial time over techniques, we found significant differences between *GravityZone+* and *DiscPIM* ($t_{23} = -8.27$, $p < 0.001$), *GravityZone+* and *CylinderPIM* ($t_{23} = -3.40$, $p < 0.05$), as well as *DiscPIM* and *CylinderPIM* ($t_{23} = -3.13$, $p < 0.05$). The difference between *GravityZone+* and *ConeExpand* was not significant ($t_{23} = -2.44$, $p = 0.136$), between *CylinderPIM* and *ConeExpand* was not significant ($t_{23} = 0.0586$, $p = 1.0$), and between *DiscPIM* and *ConeExpand* was not significant ($t_{23} = -2.05$, $p = 0.31$). Search trial time was significantly different between High and Low ($t_{23} = 5.48$, $p < .001$) DENSITY levels, also High and Medium ($t_{46} = 4.15$, $p = 0.001$), however Low and Medium levels were not significantly different ($t_{23} = -1.83$, $p = 0.24$). The interaction effect of TECHNIQUE×DENSITY was not significant ($F_{6,138} = 1.123$, $p = 0.352$).

As for the repeat time, the effect of TECHNIQUE was significant ($F_{3,69} = 18.229$, $p < .001$), and the effect of DENSITY was also significant ($F_{2,46} = 28.804$, $p < .001$). On average, repeat trials took 4.46 seconds with *DiscPIM*, 4.49 seconds with *CylinderPIM*, 4.06 seconds with *ConeExpand*, and 6.62 seconds with *GravityZone+*. For repeat trial time over techniques, we found significant differences between *GravityZone+* and *DiscPIM* ($t_{23} = 6.97$, $p < .001$), between *GravityZone+* and *CylinderPIM* ($t_{23} = 6.83$, $p < .001$), and between *GravityZone+* and *ConeExpand* ($t_{23} = 6.30$, $p < .001$). The rest of the differences between individual techniques for repeat trials were not significant: ($t_{23} = 0.215$, $p = 1.0$) for *DiscPIM* and *CylinderPIM*, ($t_{23} = 1.579$, $p = 0.78$) for *DiscPIM* and *ConeExpand*, ($t_{23} = 1.47$, $p = 0.936$) for *CylinderPIM* and *ConeExpand*. For DENSITY levels, Medium and High were significantly different ($t_{23} = 6.01$, $p < 0.001$), as well as Low and High ($t_{23} = 9.72$, $p < 0.001$). There was no significant difference between Low and Medium densities ($t_{23} = -2.26$, $p = 0.1$). There was also a significant interaction effect of TECHNIQUE×DENSITY variables on the trial time over repeat task trials ($F_{6,138} = 2.358$, $p < 0.05$).

Performance with each selection TECHNIQUE across all DENSITY levels separately is highlighted in Figure 10. To test **H3**, we ran pairwise tests on the search and repeat trials for *DiscPIM* and *CylinderPIM* on different DENSITY levels. For search trials, we found significant difference between high and low DENSITY levels ($t_{23} = 3.568$, $p < 0.01$) when *DiscPIM* was used, as well as high and low DENSITY levels ($t_{23} = 2.801$, $p < 0.05$) when *CylinderPIM* was used. For repeat trials, there was no significant difference between any of the DENSITY levels for *DiscPIM* or *CylinderPIM*. On average, participants performed the fifth trial faster than the first for search ($t_{23} = 4.868$, $p < 0.001$) and repeat ($t_{23} = 3.78$, $p < 0.001$). However, we protected against order effects by counterbalancing the TECHNIQUE order and the order of DENSITY levels.

6 DISCUSSION

The user study results indicate that our novel techniques *DiscPIM* and *CylinderPIM* perform well when compared to the benchmark selection techniques. However, we did not find a significant difference in user performance with *DiscPIM* and *ConeExpand* in neither Search nor Repeat tasks and thus we could not prove hypotheses **H1s** and **H1r**. We accept **H2s** and **H2r** because mean search times were lower for *CylinderPIM* than for *GravityZone+* for search and repeat tasks. Futher, for search tasks, DENSITY had an effect on user performance with *DiscPIM* and *CylinderPIM* in our study, thus we reject **H3s1** and **H3s2** based on our results. However, we did not find a significant effect of DENSITY level on user performance with the same techniques during repeat trials in our task, so we could not reject **H3r1** and **H3r2**. *DiscPIM* performing similarly to *ConeExpand* and *CylinderPIM* outperforming *GravityZone+* illustrate that practitioners can use either a flat or a 3D version of a mini-map that preserve the spatial relationships between objects without sacrificing object selection speed when compared to alternative techniques. However, it would be interesting to investigate if this holds true for other kinds of tasks where identifying, locating or viewing distant objects in cluttered environments is the primary goal.

We believe that such results are observed because the mini-map-based techniques preserve the spatial relationships between the objects in the environment, and given that they require more steps to select the target objects, a question arises of whether users feel more strain when using them as compared to alternatives. Figure 11 shows the SUS and TLX scores for all techniques. We ran a Friedman non-parametric test on the TLX and SUS Questionnaires data and found that SUS results were relatively high and did not show any significant differences for TECHNIQUE ($\chi^2(3) = 4.686$, $p = 0.196$). As for NASA TLX data, we found that only effort scores were significant ($\chi^2(3) = 9.132$, $p < 0.05$). Wilcoxon Signed-Ranks Tests showed that *GravityZone+* required more effort than other tech-

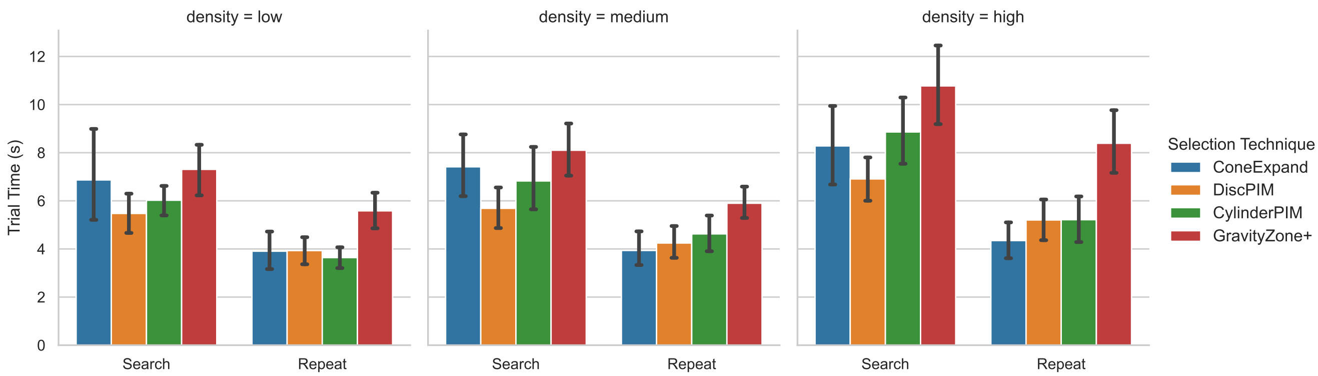

Figure 10: Average trial time in seconds for each selection TECHNIQUE across search and repeat tasks over the three DENSITY levels separately. Confidence intervals are 95%.

niques: *GravityZone+* and *CylinderPIM* ($Z = -2.104$, $p < 0.05$), *GravityZone+* and *DiscPIM* ($Z = -2.114$, $p < 0.05$), *GravityZone+* and *ConeExpand* ($Z = -2.386$, $p < 0.05$). These TLX effort results show that our techniques do not require more workload than the alternatives, despite requiring two hands and multiple steps to use.

6.1 Possible Modifications and Application Scenarios

DiscPIM and *CylinderPIM* can be modified and adapted to more environments. For example, in our scenarios, 8 slots for the circumference of the *DiscPIM* were enough to select the targets reliably, however, if more slots are required, it can be done with a trivial modification. Further, we did not experiment with adding a circumference display with slots to the *CylinderPIM*, which may be helpful in situations where multiple selectable objects overlap in the virtual environments, such as in scene editor applications. Using ray casting to select the items from *DiscPIM* is another possible modification if grabbing is not preferred.

Since our mini-map implementations follow the left controller around, this also allows contextual information to be brought from one place in an environment to another, or for use as an inventory system, for example in video games. Akin to taking a picture, our techniques capture the state of the objects at the last frame when the left trigger button was pressed, and this local representation is frozen until the trigger button is pressed again. This may potentially be used as a way to reduce memory load, though further investigation is required. In addition, both of our mini-map implementations do not fully hide what is behind them, as they are easily move-able whereas *ConeExpand* either creates a stationary grid that hides [9] or flushes [48] what is behind it. In comparison to *GravityZone+*, which modifies the original environment to make the occluded objects accessible, our mini-maps keep the environment's state intact until a selection is made. Figure 12 shows an application where a user is editing a scene with a *CylinderPIM*. This technique could also be extended to enable the insertion of objects into the environment, allowing for additional object manipulation.

Due to the ability to retain relative distances between objects in the environment, our novel techniques offer ways to visualize different kinds of data. For example, in medical simulations, anatomy students could use the mini-map-based techniques to visualize human body organs and their positions inside the body. Or in chemical visualization where areas of molecular structures can be better investigated with the relative distances between them preserved. Civil Engineers and firefighters could use our techniques to visualize city infrastructure to aid with decision making and construction.

6.2 Limitations and Future Work

We have shown that our novel techniques work well in highly occluded and dense environments, however, more evaluations can be done to test our techniques across more environments and applications. In future work we plan to compare *DiscPIM* and *CylinderPIM* to more alternative techniques that are designed for occlusion, such as *LassoGrid+* and *AlphaCursor* [48]. We also plan to include different kinds of shapes, and multiple kinds of occlusion, such as with different object sizes, and to place the participant at the center of such an environment instead of in front of it. These steps will bring more ecological validity to an experiment.

The *search and repeat* [33, 48] study design is an improvement over a simple task where a unique visible object is selected, however, an even better design can be implemented. When multiple occluded objects in a real environment look the same, accurate selection is more challenging and selection techniques that preserve the spatial relationships have a better chance to succeed. A realistic task that would test the limits of selection techniques would include selecting among objects that look the same, for example in an editor-like-scene where a user would have to select parts of a building structure. Our *CylinderPIM* can be adapted for such a use case, by enabling the adjustment of the highlighting cone size, and we plan to continue experimenting with our new techniques to test them in such scenarios. Thus, our next experiment will consist of an evaluation of our novel selection techniques compared to others from the literature in environment settings where the target will not be different from other distractors in the scene.

During the practice portion of study we relied on the participant's verbal feedback for determining their readiness, and on average participants spent around a minute and a half practicing. This is in-line with other selection studies where participants practiced for 90 seconds [27], until they were ready [9], or were familiar with the technique [48]. Despite the steps that we took to prepare the participants for completing the trials with each of the techniques, on average participants finished the trials faster as they progressed in the study. This is likely due to participants overestimating their readiness level, and since all of the techniques required multiple steps and controller/joystick manipulations for selecting the targets, some room for improvement remained as the study advanced. With each technique, over time participants figured out how to search the target occurrence area more effectively. In future work, we plan to extend the practice session to at least 3 or 4 minutes per technique and we recommend other researchers who study selection techniques for occluded environments to do the same.

Another challenge and a limitation of any research that compares selection techniques comes from the need to set the parameters for each individual technique to make a fair comparison. A distinction

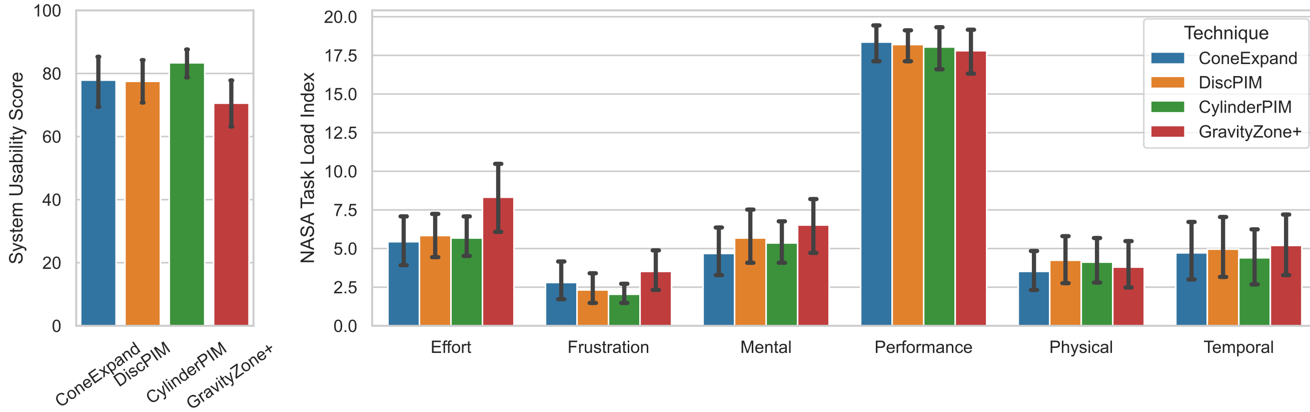

Figure 11: System Usability Scores (left), NASA TLX scores (right). Bars indicate 95% confidence intervals.

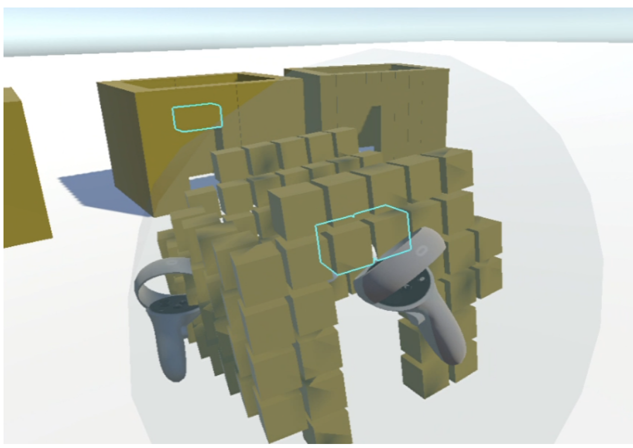

Figure 12: A user with a *CylinderPIM* selecting parts of a structure in a VR editor application.

of our *GravityZone+* implementation compared to its original implementation [48] was the use of a different joystick-tilt-to-speed curve (see Section 3.3). While we found the parameters that worked best for our task and pool of pilot participants, the ideal values are task-dependent and will change with the difference in levels of user experiences with joystick interfaces. The three remaining techniques all relied on a cone for highlighting, so to make their performance results comparable, we set their cones to exactly the same size that the pilot study participants found comfortable. A different selection task would require modifying the highlighting cone size, which could lead to different performance results.

6.3 Recommendations

Based on the design of our techniques and the evaluative scenarios we used, we can distill a set of design recommendations for the creation of selection techniques in 3D VEs characterized by different levels of occlusion and density of objects.

- *When maintaining the spatial relationships between objects is important*, we recommend using *DiscPIM* and *CylinderPIM*, with *3DMimiMap* being preferred in visualization tasks, as it provides more spatial information.
- *To provide high selection accuracy and speed* with a single unique target available, both grid-based techniques such as *LassoGrid+* and *ConeExpand*, as well as our new interactive mini-map 2D and 3D techniques are preferred.
- *To investigate large areas during selection*, we suggest using *CylinderPIM* as it displays the whole area that is highlighted with the cone and preserves all spatial relationships between objects, including depth. An additional feature of resizing the selection cone with the controller joystick may improve user experience.
- *If two hands are available* during selection in a VR environment, *CylinderPIM* or *DiscPIM* will both provide fast object selection, however when only one hand is available, either due to environment requirements or even a physical limitation, use *ConeExpand*.
- *When the user must be able to directly grab objects in their range of motion*, *DiscPIM* and *CylinderPIM* are preferable, as grabbing using the right hand is their default mode of operation. Using *GravityZone+* in such a scenario would require the user to disable and enable the technique to deactivate the threshold that makes selectable objects transparent.

7 Conclusion

In this paper, we have presented the implementation and evaluation study of two novel object selection techniques (*DiscPIM* and *CylinderPIM*) that maintain the spatial relationships between the minimized versions of the original objects. Through a user study we have shown that our techniques help quickly acquire fully-occluded objects in dense environments. We have found that our techniques outperform a previously best-performing technique (*GravityZone+*) that also preserves the object's spatial relationships. Our results suggest that future research in VR selection techniques should consider highly occluded environments to ensure that frustration-free selection methods for dense cluttered environments are developed especially for industrial applications involving remote root cause identification, maintenance and repairs.

Acknowledgments

This work is supported in part by NSF Award IIS-1917728, Northrop Grumman., Unknot.id, and the Florida High Tech Corridor Council Industry Matching Research Program. We also thank the anonymous reviewers for their insightful feedback and the ISUE lab members for their support.

References

[1] F. Argelaguet, C. Andujar, and R. Trueba. Overcoming eye-hand visibility mismatch in 3d pointing selection. In *Proceedings of the 2008 ACM Symposium on Virtual Reality Software and Technology*, VRST '08, p. 43–46. Association for Computing Machinery, New York, NY, USA, 2008. doi: 10.1145/1450579.1450588

[2] U. Assarsson, N. Elmqvist, and P. Tsigas. Image-space dynamic transparency for improved object discovery in 3d environments. Technical report, Chalmers University of Technology, Technical Report 2006-10, 2006.

[3] F. Bacim, M. Nabiyouni, and D. A. Bowman. Slice-n-swipe: A free-hand gesture user interface for 3d point cloud annotation. In *Proc. 3DUI*, pp. 185–186. IEEE, Minneapolis, Minnesota, USA, 2014. doi: 10.1109/3DUI.2014.6798882

[4] J. Bergström, T.-S. Dalsgaard, J. Alexander, and K. Hornbæk. How to Evaluate Object Selection and Manipulation in VR? Guidelines from 20 Years of Studies. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, CHI '21, pp. 1–20. Association for Computing Machinery, New York, NY, USA, May 2021. doi: 10.1145/3411764.3445193

[5] D. A. Bowman and L. F. Hodges. An evaluation of techniques for grabbing and manipulating remote objects in immersive virtual environments. In *Proceedings of the 1997 Symposium on Interactive 3D Graphics*, I3D '97, p. 35–ff. Association for Computing Machinery, New York, NY, USA, 1997. doi: 10.1145/253284.253301

[6] S. Bruckner and M. E. Groller. Exploded views for volume data. *IEEE Transactions on Visualization and Computer Graphics*, 12(5):1077–1084, November 2006. doi: 10.1109/TVCG.2006.140

[7] M. S. T. Carpendale, D. J. Cowperthwaite, and F. D. Fracchia. Distortion viewing techniques for 3-dimensional data. In *Proceedings IEEE Symposium on Information Visualization'96*, pp. 46–53. IEEE, San Francisco, CA, 1996. doi: 10.1109/INFVIS.1996.559215

[8] J. Cashion and J. J. LaViola. Poster: Dynamic adaptation of 3D selection techniques for suitability across diverse scenarios. In *2014 IEEE Symposium on 3D User Interfaces (3DUI)*, pp. 165–166. IEEE, Minneapolis, MN, Mar. 2014. doi: 10.1109/3DUI.2014.6798872

[9] J. Cashion, C. Wingrave, and J. J. LaViola Jr. Dense and Dynamic 3D Selection for Game-Based Virtual Environments. *IEEE Transactions on Visualization and Computer Graphics*, 18(4):634–642, April 2012. doi: 10.1109/TVCG.2012.40

[10] J. A. Cashion, C. Wingrave, and J. J. LaViola. Automatic 3D selection technique assignment using real-time scenario analysis. In *2013 IEEE Virtual Reality (VR)*, pp. 103–104. IEEE, Lake Buena Vista, FL, USA, March 2013. doi: 10.1109/VR.2013.6549383

[11] Y. Chen, J. Sun, Q. Xu, E. Lank, P. Irani, and W. Li. Global Scene Filtering, Exploration, and Pointing in Occluded Virtual Space. In *IFIP Conference on Human-Computer Interaction*, pp. 156–176. Springer International Publishing, Cham, 2021. doi: 10.1007/978-3-030-85607-6_11

[12] M. C. Chuah, S. F. Roth, J. Mattis, and J. Kolojejchick. Sdm: Selective dynamic manipulation of visualizations. In *Proceedings of the 8th Annual ACM Symposium on User Interface and Software Technology*, UIST '95, p. 61–70. Association for Computing Machinery, New York, NY, USA, 1995. doi: 10.1145/215585.215654

[13] C. Coffin and T. Hollerer. Interactive perspective cut-away views for general 3d scenes. In *3D User Interfaces (3DUI'06)*, pp. 25–28. IEEE, Alexandria, VA, USA, 2006. doi: 10.1109/VR.2006.88

[14] C. Correa, D. Silver, and M. Chen. Illustrative deformation for data exploration. *IEEE Transactions on Visualization and Computer Graphics*, 13(6):1320–1327, November 2007. doi: 10.1109/TVCG.2007.70565

[15] G. de Haan, M. Koutek, and F. H. Post. Intenselect: Using dynamic object rating for assisting 3d object selection. In *Proceedings of the 11th Eurographics Conference on Virtual Environments*, EGVE'05, p. 201–209. Eurographics Association, Goslar, DEU, 2005. doi: 10.2312/EGVE/IPT_EGVE2005/201-209

[16] J. Diepstraten, D. Weiskopf, and T. Ertl. Transparency in Interactive Technical Illustrations. *Computer Graphics Forum*, 21(3):317–325, May 2003. doi: 10.1111/1467-8659.t01-1-00591

[17] A. T. Duchowski and A. T. Duchowski. *Eye tracking methodology: Theory and practice*. Springer, Cham, Switzerland, June 2017. doi: 10.1007/978-3-319-57883-5

[18] N. Elmqvist. Balloonprobe: Reducing occlusion in 3d using interactive space distortion. In *Proceedings of the ACM Symposium on Virtual Reality Software and Technology*, VRST '05, p. 134–137. Association for Computing Machinery, New York, NY, USA, 2005. doi: 10.1145/1101616.1101643

[19] N. Elmqvist and P. Tsigas. A Taxonomy of 3D Occlusion Management Techniques. In *2007 IEEE Virtual Reality Conference*, pp. 51–58. IEEE, Charlotte, NC, USA, March 2007. doi: 10.1109/VR.2007.352463

[20] N. Elmqvist and M. E. Tudoreanu. Evaluating the effectiveness of occlusion reduction techniques for 3d virtual environments. In *Proceedings of the ACM Symposium on Virtual Reality Software and Technology*, VRST '06, p. 9–18. Association for Computing Machinery, New York, NY, USA, 2006. doi: 10.1145/1180495.1180500

[21] N. Elmqvist and M. E. Tudoreanu. Occlusion management in immersive and desktop 3d virtual environments: Theory and evaluation. *Int. J. Virtual Real.*, 6(2):21–32, 2007.

[22] T. T. Elvins, D. R. Nadeau, and D. Kirsh. Worldlets—3d thumbnails for wayfinding in virtual environments. In *Proceedings of the 10th Annual ACM Symposium on User Interface Software and Technology*, UIST '97, p. 21–30. Association for Computing Machinery, New York, NY, USA, 1997. doi: 10.1145/263407.263504

[23] A. Forsberg, K. Herndon, and R. Zeleznik. Aperture based selection for immersive virtual environments. In *Proceedings of the 9th annual ACM symposium on User interface software and technology*, UIST '96, pp. 95–96. Association for Computing Machinery, New York, NY, USA, Nov. 1996. doi: 10.1145/237091.237105

[24] S. Fukatsu, Y. Kitamura, T. Masaki, and F. Kishino. Intuitive control of "bird's eye" overview images for navigation in an enormous virtual environment. In *Proceedings of the ACM symposium on Virtual reality software and technology*, VRST '98, pp. 67–76. Association for Computing Machinery, New York, NY, USA, 1998. doi: 10.1145/293701.293710

[25] T. Grossman and R. Balakrishnan. The bubble cursor: enhancing target acquisition by dynamic resizing of the cursor's activation area. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '05, pp. 281–290. Association for Computing Machinery, New York, NY, USA, Apr. 2005. doi: 10.1145/1054972.1055012

[26] D. Kalkofen, E. Veas, S. Zollmann, M. Steinberger, and D. Schmalstieg. Adaptive ghosted views for augmented reality. In *2013 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pp. 1–9. IEEE, Adelaide, SA, Australia, Oct 2013. doi: 10.1109/ISMAR.2013.6671758

[27] R. Kopper, F. Bacim, and D. A. Bowman. Rapid and accurate 3D selection by progressive refinement. In *2011 IEEE Symposium on 3D User Interfaces (3DUI)*, pp. 67–74. Singapore, Mar. 2011. doi: 10.1109/3DUI.2011.5759219

[28] J. J. LaViola Jr, E. Kruijff, R. P. McMahan, D. Bowman, and I. P. Poupyrev. *3D user interfaces: theory and practice*. Addison-Wesley Professional, 2017.

[29] J. Liang and M. Green. Jdcad: A highly interactive 3d modeling system. *Computers & Graphics*, 18(4):499–506, 1994. doi: 10.1016/0097-8493(94)90062-0

[30] M. J. McGuffin, L. Tancau, and R. Balakrishnan. Using deformations for browsing volumetric data. In *IEEE Visualization, 2003. VIS 2003*, pp. 401–408. IEEE, Seattle, WA, USA, 2003. doi: 10.1109/VISUAL.2003.1250400

[31] D. Mendes, D. Medeiros, E. Cordeiro, M. Sousa, A. Ferreira, and J. Jorge. PRECIOUS! Out-of-reach selection using iterative refinement in VR. In *2017 IEEE Symposium on 3D User Interfaces (3DUI)*, pp. 237–238. IEEE, Los Angeles, CA, USA, Mar. 2017. doi: 10.1109/3DUI.2017.7893359

[32] R. A. Montano-Murillo, C. Nguyen, R. H. Kazi, S. Subramanian, S. DiVerdi, and D. Martinez-Plasencia. Slicing-Volume: Hybrid 3D/2D Multi-target Selection Technique for Dense Virtual Environments. In *2020 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 53–62. IEEE, Atlanta, GA, USA, March 2020. doi: 10.1109/VR46266.2020.00023

[33] J. Petford, M. A. Nacenta, and C. Gutwin. Pointing All Around You: Selection Performance of Mouse and Ray-Cast Pointing in Full-Coverage Displays. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, CHI '18, pp. 1–14. Association for Computing Machinery, New York, NY, USA, Apr. 2018. doi: 10.1145/3173574.3174107

[34] V. Popescu, P. Rosen, and N. Adamo-Villani. The graph camera. In *ACM SIGGRAPH Asia 2009 papers*, SIGGRAPH Asia '09, pp. 1–8.

Association for Computing Machinery, New York, NY, USA, Dec. 2009. doi: 10.1145/1661412.1618504
[35] I. Poupyrev, M. Billinghurst, S. Weghorst, and T. Ichikawa. The go-go interaction technique: Non-linear mapping for direct manipulation in vr. In *Proceedings of the 9th Annual ACM Symposium on User Interface Software and Technology*, UIST '96, p. 79–80. Association for Computing Machinery, New York, NY, USA, 1996. doi: 10.1145/237091.237102
[36] G. Ramos, G. Robertson, M. Czerwinski, D. Tan, P. Baudisch, K. Hinckley, and M. Agrawala. Tumble! splat! helping users access and manipulate occluded content in 2d drawings. In *Proceedings of the Working Conference on Advanced Visual Interfaces*, AVI '06, p. 428–435. Association for Computing Machinery, New York, NY, USA, 2006. doi: 10.1145/1133265.1133351
[37] J. Schjerlund, K. Hornbæk, and J. Bergström. Ninja hands: Using many hands to improve target selection in vr. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, CHI '21. Association for Computing Machinery, New York, NY, USA, 2021. doi: 10.1145/3411764.3445759
[38] H. Sonnet, S. Carpendale, and T. Strothotte. Integrating expanding annotations with a 3d explosion probe. In *Proceedings of the Working Conference on Advanced Visual Interfaces*, AVI '04, p. 63–70. Association for Computing Machinery, New York, NY, USA, 2004. doi: 10.1145/989863.989871
[39] R. W. Soukoreff and I. S. MacKenzie. Towards a standard for pointing device evaluation, perspectives on 27 years of fitts' law research in hci. *Int. J. Hum.-Comput. Stud.*, 61(6):751–789, December 2004. doi: 10.1016/j.ijhcs.2004.09.001
[40] F. Steinicke, T. Ropinski, and K. Hinrichs. *Object Selection In Virtual Environments Using an Improved Virtual Pointing Metaphor*, p. 320–326. Springer Netherlands, Dordrecht, 2006. doi: 10.1007/1-4020-4179-9_46
[41] R. Stoakley, M. J. Conway, and R. Pausch. Virtual reality on a wim: Interactive worlds in miniature. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '95, p. 265–272. ACM Press/Addison-Wesley Publishing Co., USA, 1995. doi: 10.1145/223904.223938
[42] J. Sun and W. Stuerzlinger. Selecting and sliding hidden objects in 3d desktop environments. In *Proceedings of the 45th Graphics Interface Conference on Proceedings of Graphics Interface 2019*, GI'19, pp. 1–8. Canadian Human-Computer Communications Society, Waterloo, CAN, 2019. doi: 10.20380/GI2019.08
[43] L. Vanacken, T. Grossman, and K. Coninx. Exploring the Effects of Environment Density and Target Visibility on Object Selection in 3D Virtual Environments. In *2007 IEEE Symposium on 3D User Interfaces*. IEEE, Charlotte, NC, USA, Mar. 2007. doi: 10.1109/3DUI.2007.340783
[44] L. Wang, J. Chen, Q. Ma, and V. Popescu. Disocclusion Headlight for Selection Assistance in VR. In *2021 IEEE Virtual Reality and 3D User Interfaces (VR)*, pp. 216–225. Lisboa, Portugal, Mar. 2021. doi: 10.1109/VR50410.2021.00043
[45] Y. Wang and R. Kopper. Efficient and Accurate Object 3D Selection With Eye Tracking-Based Progressive Refinement. *Frontiers in Virtual Reality*, 2, June 2021. doi: 10.3389/frvir.2021.607165
[46] J. O. Wobbrock, L. Findlater, D. Gergle, and J. J. Higgins. The aligned rank transform for nonparametric factorial analyses using only anova procedures. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '11, p. 143–146. Association for Computing Machinery, New York, NY, USA, 2011. doi: 10.1145/1978942.1978963
[47] D. Yu, H.-N. Liang, K. Fan, H. Zhang, C. Fleming, and K. Papangelis. Design and Evaluation of Visualization Techniques of Off-Screen and Occluded Targets in Virtual Reality Environments. *IEEE Transactions on Visualization and Computer Graphics*, 26(9):2762–2774, Sept. 2020. doi: 10.1109/TVCG.2019.2905580
[48] D. Yu, Q. Zhou, J. Newn, T. Dingler, E. Velloso, and J. Goncalves. Fully-Occluded Target Selection in Virtual Reality. *IEEE Transactions on Visualization and Computer Graphics*, 26(12):3402–3413, Dec. 2020. doi: 10.1109/TVCG.2020.3023606