PwP: Permutating with Probability for Efficient Group Selection in VR

Jian Wu, Weicheng Zhang, Handong Chen, Wei Lin, Xuehuai Shi, and Lili Wang

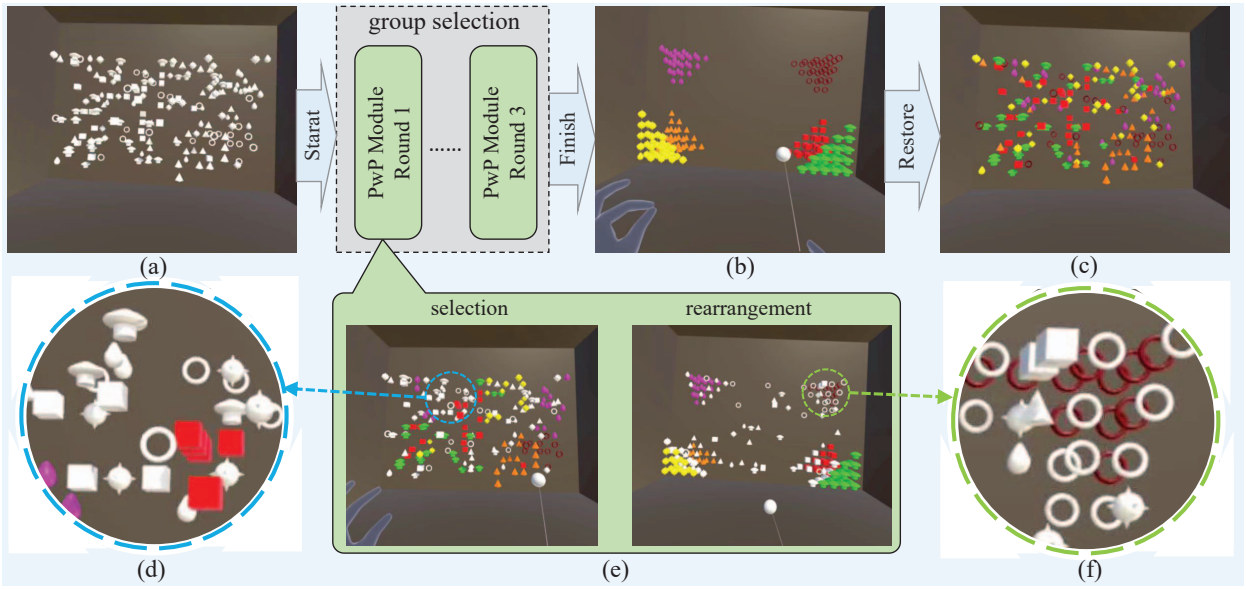


Fig. 1: An example of the PwP group selection process. For a scene with many objects scattered (a), a user can use our method to select these objects. (b) shows the result of group-selected objects, and each group is marked with a specific color. Finally, by restoring them, all the objects can be placed where they were originally (c). Our group selection method consists of multiple rounds of the PwP module, which contains a selection and rearrangement submodules (e). In the selection submodule, the user can mark several objects with a color to group-select them, and in the rearrangement submodule, a probability-based rearrangement algorithm is applied to change the object's layout to facilitate more batch selection. Through the PwP module, the objects can be rearranged from a disorganized layout (d) to a well-ordered one (f).

Abstract—Group selection in virtual reality is an important means of multi-object selection, which allows users to quickly group multiple objects and can significantly improve the operation efficiency of multiple types of objects. In this paper, we propose a group selection method based on multiple rounds of probability permutation, in which the efficiency of group selection is substantially improved by making the object layout of the next round easier to be batch-selected through interactive selection, object grouping probability computation, and position rearrangement in each round of the selection process. We conducted ablation experiments to determine the algorithm coefficients and validate the effectiveness of the algorithm. In addition, an empirical user study was conducted to evaluate the ability of our method to significantly improve the efficiency of the group selection task in an immersive virtual reality environment. The reduced operations also indirectly reduce the user task load and improve usability.

Index Terms—Group selection, permutation, probability propagation, user study, virtual reality

1 Introduction

The interaction between users and virtual objects is at the core of creating immersive interactive experiences in virtual reality (VR) environments. Object selection, as one of the hot topics in this field, is a critical component. Currently, researchers have proposed various selection methods [5, 14, 43, 45], including single-object and multi-object selection, and have achieved good results in improving selection accuracy and efficiency. However, in complex virtual scenes where many diverse objects are randomly scattered, there is often a need to quickly classify and select different types of virtual objects. To this end, it is necessary to develop a method that can rapidly select and group many randomly distributed and diverse objects. For instance, in VR strategy games, players need to quickly select and organize multiple units on the battlefield to address complex tactical needs. In VR home design, users must efficiently select and arrange various furniture pieces. In VR content creation applications, artists can swiftly select and categorize 3D parts using group selection methods, enhancing the efficiency of tasks like coloring large numbers of components.

Existing virtual object selection methods face several challenges when applied to the group selection scenarios described above. In highly cluttered multi-object scenes, objects within the same group are seldom clustered together, making it difficult to enhance efficiency

- *Jian Wu, Weichen Zhang, Handong Chen, and Wei Lin are with the State Key Laboratory of Virtual Reality Technology and Systems, Beihang University, Beijing, China. Email: lanayawj@buaa.edu.cn, summer_05_05@163.com, 22371245@buaa.edu.cn, 22371336@buaa.edu.cn.*
- *Xuehuai Shi is with Nanjing University of Posts and Telecommunications. Email: xuehuai@njupt.edu.cn.*
- *Lili Wang is with State Key Laboratory of Virtual Reality Technology and Systems, Beihang University, Beijing, China; Peng Cheng Laboratory, Shenzhen, China. Lili Wang is the corresponding author. Email: wanglily@buaa.edu.cn.*

Manuscript received xx xxx. 201x; accepted xx xxx. 201x. Date of Publication xx xxx. 201x; date of current version xx xxx. 201x. For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org. Digital Object Identifier: xx.xxxx/TVCG.201x.xxxxxxx

through box or volume selection techniques, especially when dealing with large numbers of objects. Additionally, some multi-ray or multi-bubble selection methods require precise operations, which can quickly lead to a heavy workload for users.

To address these problems, we propose a permutating with probability (PwP) method to facilitate efficient multi-object group selection in an immersive virtual environment. The core principle of this method is to update the object layout based on grouping probabilities, thereby reducing randomness and enabling more efficient batch selections in subsequent rounds. The PwP method calculates the group probabilities for all objects to be selected based on the user's group selection input. It then rearranges the positions of all objects according to these probability calculations to create a layout that is optimized for batch selection, ultimately improving selection efficiency. To achieve this, we introduce the probability propagation and position rearrangement algorithms for probabilistic permutation. The probability propagation algorithm begins with the objects that have been selected by the group, then calculates and updates the grouping probabilities of unselected objects following specific rules (e.g., decay with distance, fluctuations). Subsequently, the objects are repositioned near their most likely group center based on their updated probabilities, leading to a more structured and efficient layout. We conducted an empirical user study to evaluate the performance of our PwP method. Compared to the traditional and the state-of-the-art methods, our method performs best under conditions of varying group sizes and levels of layout orderedness. In addition, our approach significantly reduces the task load and improves the usability of group selection tasks. Fig. 1 shows an example of our PwP method workflow. After several (in this case, 3) rounds of the PwP module, originally scattered objects are all selected and grouped. Finally, one can choose to restore the objects to their original positions.

The contributions of this paper are summarised as follows:

- We proposed a permutating with probability multi-object group selection framework, which allows multiple rounds of interactive selection and rearrangement of target objects to enable dramatic efficiency improvement.
- We develop a probability propagation algorithm based on two strategies to update the grouping probability to decide the grouping priority of the remaining objects to be selected.
- We introduced a position rearrangement algorithm based on the grouping probability to relocate the positions of the target objects, which can reduce the layout randomness and make the next selection round easier.
- We evaluated the performance of our proposed PwP method through ablation experiments and conducted an empirical user study to validate the method's effectiveness in an immersive virtual environment.

In summary, our approach combines advanced algorithms and user-centered design to provide an innovative solution to the group selection problem in VR environments. These innovations not only advance the development of VR interaction technology but also open up new possibilities for future research and applications.

2 RELATED WORK

We briefly introduce the prior works from three perspectives: single-object selection, multi-object selection, and object group selection. For a more comprehensive illustration, we recommend readers to the survey papers [1] and [22].

2.1 Single-object selection

Single-object selection techniques allow the user to pick one virtual object per selection operation [2, 4, 5, 7, 14, 18, 41, 43, 46, 47]. Early single-object selection techniques were primarily based on emitting rays, which were calculated to intersect objects and confirm the target object selected by the user [5, 12, 15, 46]. For example, Olwal et al. present a virtual, flexible pointer that allows a user in a 3D environment to point more easily to fully or partially obscured objects [15]. Later, a dynamic object rating method was proposed to assist 3D object selection and increase selecting accuracy [12]. To increase the usability of selection techniques, some researchers focus on the selection in dense environments [7, 39]. Recently, Delamare et al. proposed MultiFingerBubble, a new variation of the 3D Bubble Cursor, which can select multiple target objects with the user's multiple fingers [14].

In addition to ray-based methods, many object manipulation methods also support the ability to select object [9, 21, 26–28, 37, 44]. Early explorations of this type of research propose novel and efficient object manipulation methods, like Go-Go [28], WIM [37], and Voodoo Dolls [26]. In recent years, researchers have proposed many optimization methods based on them. Wu et al. proposed EEBA, an efficient and ergonomic big-arm for distant object manipulation based on elbow angle mapping optimization [44]. The task of these methods is object manipulation. However, objects must be selected before they can be manipulated, so they also enable object selection.

The above methods are mainly conducted using VR controllers. With the development of computer vision and multi-modal technology, more and more selection methods are using gestures, head movement, eye movements, and other input methods [3, 8, 10, 20, 24, 32–35, 40, 42]. Sidenmark et al. introduced Outline Pursuits, which extends a primary pointing modality for gaze-assisted selection of occluded objects [33]. They have conducted a user study to compare gaze, head, and controller-based selection [35]. Luong et al. investigated the performance differences between controller and bare-hand-based interactions in VR [20].

2.2 Multi-object selection

There are already several works on multi-object selection, which allow users to select multiple objects at a time [19, 31, 45, 49, 51]. Zhang et al. presented a new 3D user interface for selecting an arbitrarily shaped region of interest (ROI) in virtual reality [49]. Wu et al. compared point-based selection and volume-based batch selection techniques through a user study [45]. The results showed that the point-based technique was more efficient and robust than the volume-based techniques in dense and disorganized environments.

Another research direction focuses on regional multi-object selection in high-density environment, e.g., point cloud selection, protein molecule region selection, etc. [23, 29, 36, 48, 50] In the earlier decade, Stenholt et al. explored various selection tools, such as a brush, a lasso, and a magic wand for large-scale multi-object selection [36]. Later, Yu et al. presented a family of three interactive Context-Aware Selection Techniques (CAST) for analyzing large 3D particle datasets, which improves usability and speed of spatial selection in point clouds [48]. Recently, Zhao et al. proposed three novel spatial data selection techniques that are context-aware and be suitable for a wide range of data features and complex scenarios [50].

2.3 Object group selection

Group selection was first studied in a two-dimensional screen virtual environment for 3D modeling applications [25, 38]. The groups of objects in these works are predefined, such as gravitational hierarchy. Debarba proposed the LOP cursor, a metaphor that shows possible suitability for group selection with its two-legged cursor design (one for selection and the other for grouping) [13]. Unlike the broad exploration of object selection techniques, little prior work has been studied on the grouping selection for immersive virtual environments. Some researchers conducted later alignment tasks for grouped objects. For example, Felice et al. created StickyLines, a tool that treats guidelines for object alignment [11]. Shi et al. present four interaction techniques for three degrees-of-freedom translational alignments [30].

In our paper, the object group selection task refers to the technology that allows users to freely group scene objects through an interactive mode of object selection and classification. Unlike multi-object selection, which focuses on selecting multiple objects simultaneously, group selection emphasizes the speed and ease of selecting and classifying scene objects. Group selection is a composite task that combines object selection techniques with label classification methods. However, group selection can be inefficient in scenes with a large number of objects

This article has been accepted for publication in IEEE Transactions on Visualization and Computer Graphics. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TVCG.2025.3549560

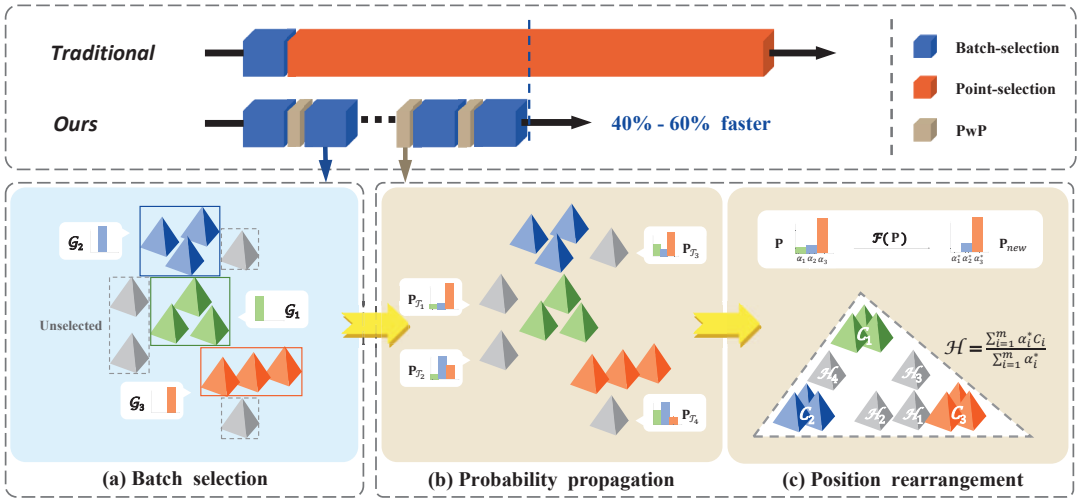


Fig. 2: Pipeline of our PwP group selection method. Compared to the traditional batch and point selection approach, we facilitate multiple rounds of batch selection with the PwP module consisting of two stages: probability propagation and position rearrangement. During the probability propagation stage, we compute the likelihood of each object belonging to each group. In the position rearrangement stage, we then reposition all the objects based on these calculated probabilities, moving each object to a location closer to its highest grouping probability. Our group selection method enables 40-60% of efficiency improvement.

and a chaotic distribution. This paper introduces new strategies and optimization algorithms designed to increase the proportion of multi-object selection operations, thereby improving the efficiency of group selection by reducing the level of disorder in the spatial distribution of objects.

3 Method

As mentioned above, the advantage of point-and-click selection lies in its convenience for selecting a single object. Still, it is inefficient since only one object can be selected at a time. Batch selection allows for the selection of multiple objects in a single operation, which is efficient; however, it is prone to mis-grouping randomly distributed objects, making it difficult to distinguish between different groups of objects. Therefore, we propose the Permutating with Probability (PwP) method to achieve the selection of multiple groups of randomly distributed objects with fewer operations.

This section describes the details of our PwP method. The basic principle of the method involves multiple rounds of selection of objects randomly distributed in a virtual environment. After each selection round, the probability of grouping for each object is calculated, and the positions are rearranged to reduce the randomness, facilitating more batch selections in the next round. We first introduce the basic pipeline of our PwP method (Sec. 3.1), followed by detailed explanations of two important algorithms involved in the framework. Sec. 3.2 describes how to calculate the grouping probability of objects after the user completes a round of selection, and Sec. 3.3 describes how to rearrange the objects based on the calculated grouping probabilities to reduce the randomness of the objects.

3.1 Pipeline

Fig. 2 presents the pipeline of our PwP method. Unlike traditional approaches that combine batch selection with point selection for multi-object grouping, our method aims to enable the user to perform more batch selections, thereby improving the efficiency of group selection (Fig. 2, top). Our method achieves this through multiple rounds of batch selection combined with the PwP module.

Fig. 2, bottom provides a detailed explanation of the PwP module. For any selection round, the user first batch selects several objects and marks them into their groups (a). Then, the PwP module is applied. The module consists of two stages. In the first stage (b), the grouping probability of each object will be calculated. The selected objects' grouping probability will be initialized according to their groups, and others' grouping probability will be calculated according to their spatial relations to the selected objects, using our probability propagation strategy. In the second stage (c), the positions of all the objects will be rearranged according to their grouping probabilities. This stage places the selected objects into their group areas and moves un-selected objects toward their likely group areas. Once the rearrangement stage is completed, the next selection round begins, iterating until all objects are successfully selected and grouped.

3.2 Probability Propagation

Our method guides the update of object positions by assigning grouping probabilities to all objects. The goal is to arrange the objects in a way that facilitates more batch selection in the next round. In this section, we detail our strategy for calculating grouping probabilities.

3.2.1 Problem Formulation

Without loss of generality, consider that we are selecting n objects to be grouped into m groups. The set of n objects is represented as $\mathbf{T} = \{\mathscr{T}_1, \mathscr{T}_2, ..., \mathscr{T}_n\}$, and the m groups are represented as $\mathbf{G} = \{\mathscr{G}_1, \mathscr{G}_2, ..., \mathscr{G}_m\}$. For any object $\mathscr{T}_i$, we define its grouping probability as follows:

$$\mathscr{P}_{\mathscr{T}_i} = \{P(\mathscr{T}_i, \mathscr{G}_1), P(\mathscr{T}_i, \mathscr{G}_2), ..., P(\mathscr{T}_i, \mathscr{G}_m)\} \tag{1}$$

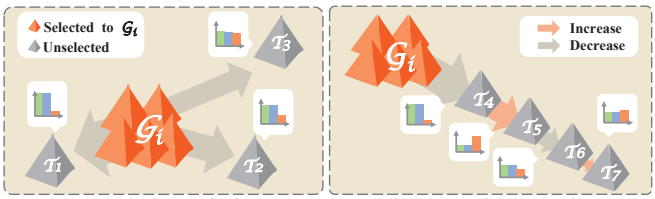
here, $P(\mathscr{T}_i, \mathscr{G}_i)$ represents the probability that object $\mathscr{T}_i$ should be selected into group $\mathscr{G}_i$. For all objects, the grouping selection probability matrix is an $n \times m$ matrix.

$$\mathbf{P} = \{\mathscr{P}_{\mathscr{T}_1}^\top, \mathscr{P}_{\mathscr{T}_2}^\top, ..., \mathscr{P}_{\mathscr{T}_n}^\top\}^\top \tag{2}$$

Thus, the goal of the group selection task is to optimize the row vectors of this matrix into standard unit probability vectors containing only 0 and 1, meaning that all objects have been selected into their groups.

3.2.2 Propagation Strategy

We proposed a probability propagation algorithm to calculate the grouping probability for the objects. As shown in Fig. 2, the input contains selected and un-selected objects for our PwP module. For a selected object, since the user assigns its group, the corresponding probability element of the group in the grouping probability vector is set to 1, and others are set to 0. For an un-selected object, we define two strategies, *Proximity exclusion* and *Interval transfer*, to calculate its grouping probability.



(a) Proximity exclusion (b) Interval propagation

Fig. 3: Illustration of two propagation strategies: (a) proximity exclusion and (b) interval propagation.

Proximity exclusion We observed that after a user makes a batch selection, objects adjacent to the selected object boundaries tend to belong to different groups. Therefore, the probability that these objects belong to the selected object group should be low. As shown in Fig. 3, (a), for a set of grouped objects $\mathcal{G}_i$(represented as orange tetrahedrons), there are three adjacent un-selected objects $\mathcal{T}_1, \mathcal{T}_2, \mathcal{T}_3$,(gray tetrahedrons). According to Our *proximity exclusion* strategy, the probability of the three adjacent surrounding objects grouped into $\mathcal{G}_i$ should be low, and the closer the object is to $\mathcal{G}_i$, the lower the probability. Hence, $P(\mathcal{T}_1, \mathcal{G}_i) < P(\mathcal{T}_2, \mathcal{G}_i) < P(\mathcal{T}_3, \mathcal{G}_i)$.

Interval propagation The interval propagation strategy means that for objects that have been grouped, the probability that the object closest to it is in the same group decreases. In contrast, the probability that the object further down is in the same group increases relatively, and so on. As shown in Fig. 3, (b), for the grouped objects, there are four un-selected objects in the near. For $\mathcal{T}_4$, the probability should be low according to the *proximity exclusion* strategy. For $\mathcal{T}_5$, the probability should rise sharply, mainly affected by $\mathcal{T}_4$. For $\mathcal{T}_6$, the probability should drop normally, mainly affected by $\mathcal{T}_5$. For $\mathcal{T}_7$, the probability should rise slightly, mainly affected by $\mathcal{T}_6$. In summary, the probability from near to far should decrease and rise in turn. Moreover, the influence of such interval propagation should be limited to $\mathcal{G}_i$'s certain extent, and it will no longer be affected by $\mathcal{G}_i$ beyond a certain distance threshold.

3.2.3 Implementation

We designed a probability propagation algorithm based on the breadth-first search algorithm to calculate and update the probabilities for each object. Instead of building a search graph for the objects, which is time-consuming, we construct a spatial grid to accelerate the propagation process. We first calculate the total bounding box of n objects to be selected to construct the spatial grid. Then, we find the smallest object bounding box edge length l_{min} and set it to the edge length of the grid cube cell, and then the grid is successfully constructed, as shown in Fig. 4. This ensures that each grid cube contains at most one object.

The probability propagation algorithm mainly contains four steps: *initializing*, *queue construction*, *propagation*, and *probability amplifier*.

I. Initializing: In the initializing step, we mark the selected and grouped objects and set their grouping probability as a standard unit vector $\mathbf{e}_k$, k is the group number it belongs to. The grouping probability vector of these objects has not been updated ever since.

II. Queue construction: The queue construction process is similar to the Breadth-First Search algorithm. Fig. 4, (b) illustrates the queue construction (top) and propagation process (bottom). The left is the spatial grid, and the objects $\{A,B,C,D,E\}$ are selected and grouped. They are separated into groups 1 for $A,B,andC$ and 2 for D,E, then pushed into the queue $\mathbf{Q}$ head. In this context, we call them seeds.

We apply a breadth-first search algorithm starting from the grids in the group regions. For each group region, the outer first-layer grid is tested. If an unselected object is present, it is pushed into the queue. Then, we apply the same test to the second-layer grid until all unselected objects are in the queue. For example, in Fig. 4, (b) top, objects $\{\mathcal{T}_1, \mathcal{T}_2, \mathcal{T}_3\}$ and $\{\mathcal{T}_4, \mathcal{T}_5, \mathcal{T}_6\}$ are in the first and second layers of the group 1 region. Objects $\{\mathcal{T}_7, \mathcal{T}_8\}$ and $\{\mathcal{T}_9, \mathcal{T}_{10}, \mathcal{T}_{11}\}$ are in the first and second layers of the group 2 region. Following the principles above, they are all pushed into the queue $\mathbf{Q}$ with the present order. Thus, the

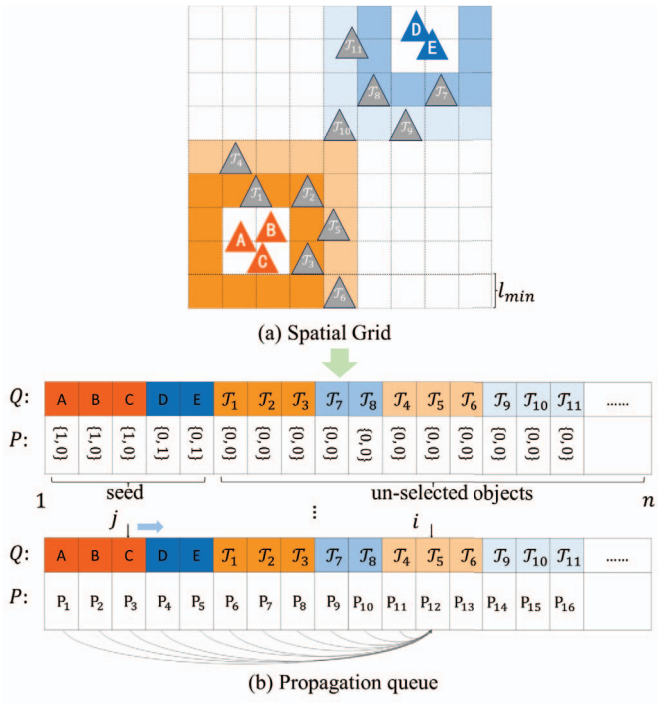


Fig. 4: Probability Propagation implementation. (a) The total bounding box of all objects is first partitioned into grid cube cells with the cube side lengths set to l_{min}. (b) Then, a propagation queue is constructed to help calculate each object's grouping probability. For an object $\mathcal{T}_i$, its grouping probability is based on all the objects in front of it.

selected and grouped objects have the highest orders, and the closer layer grid the un-selected objects are in, the higher their orders are in the queue.

III. Propagation: Since we have constructed the queue, we can apply the probability propagation along the queue. The propagation starts from the first un-selected object, e.g., object $\mathcal{T}_1$ in Fig. 4, (b) bottom. To satisfy the two propagation strategies, we introduce the calculation of the probability of each object of the propagation process as follows:

$$P_i = P_i \odot (\vec{\mathbf{1}} - \frac{P_1}{\sigma_{i1}}) \odot \cdots \odot (\vec{\mathbf{1}} - \frac{P_j}{\sigma_{ij}}), \quad j = i-1 \tag{3}$$

here, i represents the queue pointer, thus the current calculating object queue number. P_i represents the grouping probability of the i-th object in the queue. The operator $\odot$ represents the Hadamard product. σ_{ij} reflects the distance between the i-th and j-th objects in queue, which is defined as follow:

$$\sigma_{ij} = (\frac{max(|x_i - x_j|, |y_i - y_j|, |z_i - z_j|)}{l_{min}})^{\lambda} \tag{4}$$

here, (x_i, y_i, z_i) is the i-th object's position coordinate in virtual environment. λ is a user-defined coefficient, we set $\lambda = 0.3$ in this paper.

The essence of the probability propagation is the design of the distance descriptor σ_{ij} in Eq. (4). On the one hand, σ_{ij} increase when the distance of i-th and j-th objects increase, and $\vec{\mathbf{1}} - \frac{P_j}{\sigma_{ij}}$ increase, subsequently, P_i increase. This reflects the proximity exclusion strategy. On the other hand, according to the above analysis, σ_{ij} makes that the closer two objects are, the greater the difference between their grouping probabilities. This reflects the interval propagation strategy.

IV. Probability Amplifier: From Eq. (3), we know that the grouping probability of the objects in the queue is calculated from the grouping probability of the objects in the queue in front of it, and the probability propagation process of the whole queue starts from the seeds.

This article has been accepted for publication in IEEE Transactions on Visualization and Computer Graphics. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TVCG.2025.3549560

However, the total amount of probability values is not large due to the small number of seeds as a proportion of all objects in each round of user selection, and the larger distance between objects causes the impact of the propagated probabilities to be further reduced.

Therefore, we designed an amplifier to amplify the probability of grouping objects after the probability propagation to ensure the propagation wave remains significant. For an arbitrary un-selected object $\mathscr{T}_i$, suppose its grouping probability is $P_{\mathscr{T}_i} = \{\alpha_1, \alpha_2, ..., \alpha_m\}$. We first find the infinite norm $\gamma = max(\alpha_1, \alpha_2, ..., \alpha_m)$, the amplifier $\mathscr{A}$ can be illustrated as follow:

$$\mathscr{A}(P_{\mathscr{T}_i}) = \begin{cases} \frac{\beta}{\gamma} P_{\mathscr{T}_i}, & \gamma < \beta \\ P_{\mathscr{T}_i}, & \gamma \geq \beta \end{cases} \tag{5}$$

Here, we set the amplifying coefficient $\beta = 0.8$ in this paper. Note that the sum of the elements in the grouping probability vector can be larger than 1 in the context.

In addition, we also designed an iterative strategy for each round of the PwP module, which is to repeat the propagation multiple times according to the proportion of seed objects. Suppose that the proportion of selected objects for the current round in the scene is η. We can calculate the iteration times Ω as follows:

$$\Omega = \begin{cases} 4, & 0 < \eta \leq 1/8 \\ 3, & 1/8 < \eta \leq 1/4 \\ 2, & 1/4 < \eta \leq 1/2 \\ 1, & 1/2 < \eta \leq 1 \end{cases} \tag{6}$$

3.3 Position rearrangement

In Sec. 3.2, we computed the grouping probabilities of the objects with the probability propagation algorithm, and this section describes how to rearrange the objects based on the obtained grouping probabilities for the next round of user selection.

3.3.1 Rearrangement Principles

We formulate the following principles for the object rearrangement to facilitate easy batch selection for the net round user selection.

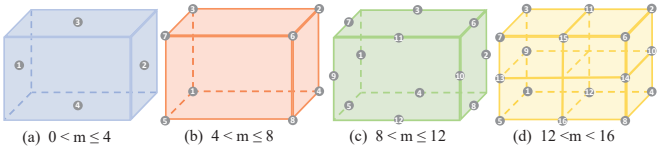
- **Decentralized group centers** Before the first round of object rearrangement, the center positions of each group need to be determined. These center positions should be as far away from each other as possible, and at the same time, try to avoid the three group center positions sharing the same line to make the subsequent selection of objects easier for users and reduce the occurrence of mis-selections.
- **Negative Correlation** This principle means that the higher the probability value of an object in a group, the closer it should be located to that group, making it more likely to be selected in the next selection round.
- **No overlapping** There should be no large overlap area between objects while ensuring that they are relocated according to their grouping probabilities.

3.3.2 Setting Group Centers

Following the first rearrangement principle, we designed several group center layout settings. We tend to set the centers on the boundary of the bounding box of all objects. As shown in Fig. 5, we design four layouts for the object rearrangement with group number m ranging from 1 to 16. The four group center layouts are used for the four group number scenarios, $0 < m \leq 4$, $4 < m \leq 8$, $8 < m \leq 12$, and $12 < m \leq 16$, respectively. When the group number $m > 16$, users are allowed to manually specify new group centers manually.

3.3.3 Objects Rearrangement

In Sec. 3.2.3, we divided the total bounding box of all objects into a cubic grid with cube side length l_{min}. Above, we have set the grouping center position vectors of the objects $\mathbf{C} = \{\mathscr{C}_1, \mathscr{C}_2, ..., \mathscr{C}_m\}$. Now, we describe how to rearrange the objects according to their grouping probabilities in the current round.



(a) $0 < m \leq 4$ (b) $4 < m \leq 8$ (c) $8 < m \leq 12$ (d) $12 < m < 16$

Fig. 5: Four group center layouts. Group center positions are marked as gray dots with numbers

The object rearrangement consists of mainly two steps: *probability truncation* and *Interpolation*.

I. Probability Truncation: We will compute the object's rearrangement position through probability interpolation to move the object closer to the possible group center. However, since the probabilities of low-probability groupings also play a role in the interpolated positions of the objects, direct use of the original grouping probabilities makes it less efficient to rearrange the positions towards the possible group centers (i.e., those with higher probabilities). We propose probability truncation to speed up this process. By setting probability components smaller than the median to zero, the effect of less probable groupings on the rearranged positions of objects can be eliminated. For example, if the original grouping probability is $P_{\mathscr{T}_i} = \alpha = \{\alpha_1, \alpha_2, ..., \alpha_m\}$, and the new grouping probability after truncation is $\alpha^* = \{\alpha_1^*, \alpha_2^*, ..., \alpha_m^*\}$. The probability truncation can be represents as the function $\mathscr{F}$ as:

$$\alpha^* = \mathscr{F}(\alpha) \tag{7}$$

satisfying:

$$\alpha_i^* = \begin{cases} \alpha_i, & \alpha_i > med(\alpha) \\ 0, & \alpha_i \leq med(\alpha) \end{cases} \tag{8}$$

here, $med(\alpha)$ represent the median value of α.

II. Interpolation: We apply a simple interpolation to compute the rearrangement location $\mathscr{H}_i(x_i, y_i, z_i)$ for a object as follow:

$$\mathscr{H}_i = \frac{\sum_{i=1}^{m} \alpha_i^* \mathscr{C}_i}{\sum_{i=1}^{m} \alpha_i^*} \tag{9}$$

After we get the rearrangement location of an object, we find the corresponding gird cube according to it and place it directly. If another object is already inside the target cube, we find the nearest empty cube to place the object at the center through the Breadth-First Search algorithm.

4 Pilot Experiment

In this section, we applied a pilot experiment to determine the two coefficients, λ in Eq. (4) and β in Eq. (5), for our probability propagation algorithm. The performance of our method is affected by the orderedness of the objects to be selected. Therefore, we define an indicator describing the object orderedness level for subsequent experiments.

4.1 Orderedness Level

For a set of objects $\mathbf{T}$ containing n objects of m groups in the scene, we aimed to define $\mathscr{S}$ to indicate their orderedness level. The larger $\mathscr{S}$ is, the more organized they are. The calculation process of $\mathscr{S}$ is illustrated below.

Single object For any single object $\mathscr{T}_i$, let $s(\mathscr{T}_i)$ reflect the proportion of space surrounding $\mathscr{T}_i$ occupied by objects in the same group, which satisfying:

$$s(\mathscr{T}_i) = \frac{1}{8} \sum_{k=1}^{8} \varepsilon_{ik} \tag{10}$$

here, ε_{ik} indicates that whether $\mathscr{T}_k^i$ belongs to the same group as $\mathscr{T}_i$. $\mathscr{T}_k^i$ represents the object closest to $\mathscr{T}_i$ in the k_{th} octant of O_{i-xyz}, where O_{i-xyz} is a space rectangular coordinate system with $\mathscr{T}_i$ as the origin and parallel to the world coordinate system. ε_{ik} is defined as follow:

$$\varepsilon_{ik} = \begin{cases} 1, & \mathscr{T}_k^i \text{ and } \mathscr{T}_i \text{ belong to the same group} \\ 0, & \mathscr{T}_k^i \text{ and } \mathscr{T}_i \text{ belong to different groups} \end{cases} \tag{11}$$

Objects set For the object set $\mathbf{T}$, the total orderedness $\mathscr{S}$ is define as:

$$\mathscr{S} = \eta\bar{\mathscr{S}} = \frac{\eta}{n}\sum_{i=1}^{n} s(\mathscr{T}_i) \tag{12}$$

here, $\bar{\mathscr{S}} = \frac{1}{n}\sum_{i=1}^{n} s(\mathscr{T}_i)$ is the average of all the object's orderedness. η is the normalization parameter to regularize the orderedness level into range $[0,1]$.

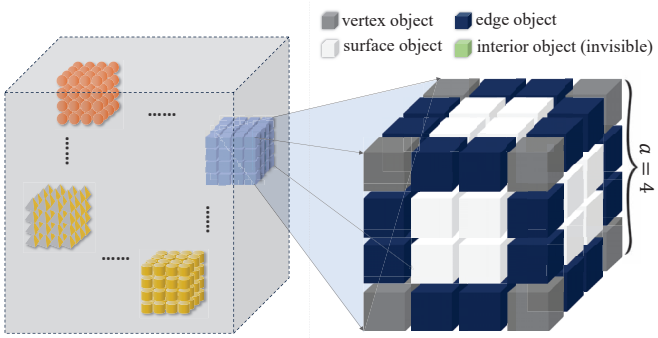
Normalization To decide the normalization parameter η, we first define two objects set states:

I. $\mathscr{S} = 0$: In this state, for each $\mathscr{T}_i$, the closest objects in the eight octants all belong to other groups.

II. $\mathscr{S} = 1$: In this state, all objects are clustered into their groups, formed as m cube sets.

As shown in Fig. 6, (a) shows the definition of the objects with $\mathscr{S} = 1$. (b) shows one of the clustered $a \times a \times a$ cube set. Each tiny cube represents an object's location. In this cube set, there are:

- $(a-2)^3$ interior objects $\mathscr{T}_{in}$, where $s(\mathscr{T}_{in}) = 1$.
- $6(a-2)^2$ white objects on surfaces $\mathscr{T}_{surf}$, where $s(\mathscr{T}_{surf}) = \frac{1}{2}$.
- $12(a-2)$ blue objects on edges $\mathscr{T}_{edge}$, where $s(\mathscr{T}_{edge}) = \frac{1}{4}$.
- 8 grey objects on vertexes $\mathscr{T}_{vert}$, where $s(\mathscr{T}_{vert}) = \frac{1}{8}$.



(a) grouped scene where $S = 1$ (b) a particular objects group

Fig. 6: (a) The objects scene with the highest orderedness level (State II). (b) One of the cube-shaped clustered groups with sidelength $a = 4$. Different colors are only used to distinguish different position types.

For the total objects set $\mathbf{T}$, the average side length $\bar{a}$ of the cube formed by each group is

$$\bar{a} = \sqrt[3]{\frac{n}{m}} \tag{13}$$

then,

$$\begin{aligned}\bar{\mathscr{S}} &= \frac{m}{n}\left[(\bar{a}-2)^3 + \frac{6(\bar{a}-2)^2}{2} + \frac{12(\bar{a}-2)}{4} + \frac{8}{8}\right] \\ &= \left(\frac{\bar{a}-1}{\bar{a}}\right)^3\end{aligned} \tag{14}$$

since in this state, we have $\mathscr{S} = 1$. Finally, we get:

$$\eta = \frac{\mathscr{S}}{\bar{\mathscr{S}}} = \frac{n}{\left(\sqrt[3]{n} - \sqrt[3]{m}\right)^3} \tag{15}$$

4.2 Coefficient Determination

In this section, we perform the pilot experiment to analyze the influence of the coefficients λ and β on our method. We designed an experiment scene (in Fig. 7) containing $n = 120$ objects in $m = 8$ groups, the orderedness level of the objects in the scene is $\mathscr{S} = 0.2$. We recruited 8 participants in the pilot experiment. They all had immersive VR application experiences before. In the experiments, they were asked to use our PwP method to group select the scattered objects in the scene following the rules below:

- Participants should make at least one and no more than one selection per group in each selection round.
- Participants shall be prioritized for batch selection.

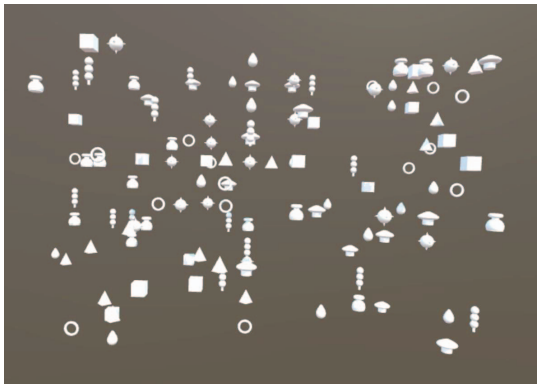


Fig. 7: The experiment scene used in the pilot user study. There were 120 objects divided into eight groups, and the different groups were distinguished by different shapes. The orderedness level is $\mathscr{S} = 0.2$.

The metric of the experiment is the number of the group selection round $\mathscr{N}_{gs}$. We first set $\lambda = 0.5$ and changed β from 0.6 to 1.0 in increments of 0.1. Then, we set $\beta = 0.8$ and changed λ from 0.2 to 0.6 in increments of 0.1. The results are shown in Tab. 1. According to the results, we set $\lambda = 0.3$ and $\beta = 0.8$ in the subsequent user study.

Table 1: The result number of the group selection round (in times), averaged within 8 participants.

β ($\lambda = 0.5$)	0.6	0.7	0.8	0.9	1.0
$\mathscr{N}_{gs}$	10.13	7.50	**6.38**	6.63	12.25
λ ($\beta = 0.8$)	0.2	0.3	0.4	0.5	0.6
$\mathscr{N}_{gs}$	6.13	**4.75**	5.50	6.38	8.25

5 User Study

5.1 Overview and Hypotheses

After the pilot experiments, we obtained the coefficients for the algorithm used in the method. In this section, we aimed to explore the efficiency performance and validate that the proposed method can reduce the number of user operations in group selection tasks. We formulated the following hypotheses:

H1. The PwP method can significantly reduce the time cost for group selection tasks.

H2. The PwP method can significantly reduce the number of user operations for group selection tasks.

5.2 Participants and Apparatus

We recruited 18 participants, ten males, and eight females (none of whom participated in the pilot experiment), ranging from 20 to 28 years old ($M = 24.39, SD = 2.11$), with normal vision (or corrected-to-normal vision by wearing glasses). Twelve had used immersive HMD VR applications before, and none reported balance disorders.

Our system used a Pico 4 pro headset powered by a workstation with a 3.8GHz Intel(R) Core(TM) i7-10700KF CPU, 32GB of RAM, and an NVIDIA GeForce RTX 3080Ti graphics card. The tracked physical space hosting the VR applications is $4 \times 4m^2$. We used Unity 2020.1 to implement our PwP group selection method. Before participants started, we measured the inter-pupillary distance (IPD) for them with a millimeter scale and adjusted the IPD of the headset to meet their best visual setting. The whole system was running at 90fps for each eye.

5.3 Study Design

We used a $3 \times 3 \times 3$ repeated measures within-subject user study design. Consider the advancement and relevance of the comparative method, we tested three group selection methods: MultiFingerBubble [14] (CC_1), point- and volume-based selection [45] (CC_2) and Our PwP method (EC), with three levels of group number $m \in \{4, 7, 10\}$, three orderedness levels $\mathscr{S} \in \{0.2, 0.5, 0.8\}$. Our PwP method supports the integration of any effective batch selection method. In this user study, to ensure consistency with CC_2, we used a cube-shaped batch selection volume. The total number of the candidate objects was the same (we set it as 120 in the study) for all conditions. For each method, each group number level, and each orderedness level, when all objects were selected and grouped, the participant completed a trial. For one group selection task, a participant should finish 27 trials. In total, we conducted $18 \times 27 = 486$ trials.

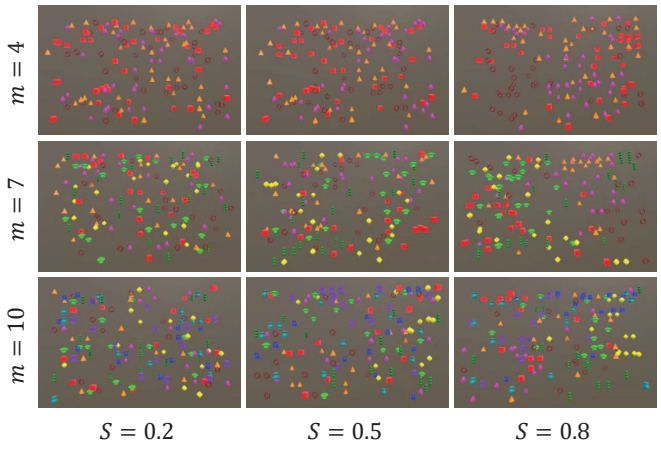


Fig. 8: Nine object layouts with different m and $\mathscr{S}$ levels were used in the user study. Here, we colored the objects for better illustration.

We conducted the user study in a virtual empty indoor room with the target objects scattered in front of the participant. Fig. 8 shows all the object layouts with different m and $\mathscr{S}$ levels. Participants were asked to select and group all presented virtual objects on each trial as quickly as possible. Objects were grouped by assigning specific colors to them with three comparison methods. The research was performed under the oversight of the Biology and Medical Ethics Committee of Beihang University, with protocol number BM20240277. Consent from the human subjects in the research was obtained.

5.4 Procedure and Metrics

The user study procedure is shown in Fig. 9. Each participant participated in the experiment for nine days. On the first day they arrived at the experiment site, they signed the information sheet. Then, we explained and showed them the operations of three group selection methods. After that, they conducted technical training to familiarize themselves with the methods. Meanwhile, we assigned each participant a specific order of the three methods. The balanced Latin square determined the order.

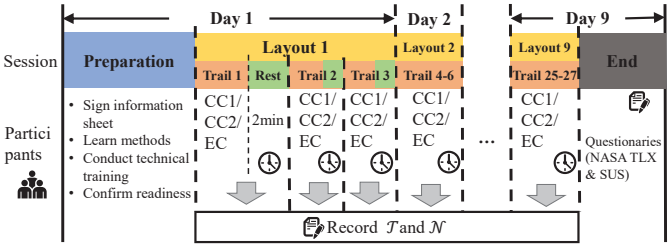


Fig. 9: The experimental procedure of the user study.

We prepared nine object layouts (with different m and $\mathscr{S}$ levels), ordered randomly for each participant throughout the nine days. When participants were confirmed to be ready, the user study began. Throughout the user study, participants were instructed to maintain their original position. While they were allowed to turn their bodies and look around, they were not permitted to walk or jump to change their position. They perform group selection tasks for each object layout using three methods in their specified order. Completing one group selection with one method is considered one trial. Participants complete three trials per day and rest for two minutes between trials. Each participant took approximately 30-60 minutes daily to complete the layout group selection task.

We counted the following two objective metrics:

- Completion time, the time it took for the participant to complete a trial.
- Number of operations, the user operation times during each trial. A complete selection and grouping are recorded as one user operation.

Despite the above metrics, we also recorded the single-object and multi-object selection times for both CC and EC methods and the number of conducting PwP modules for the EC method (marked inside the bars of the number of operations in Fig. 10).

For subjective metrics, we used the standard NASA TLX questionnaire [16, 17] to measure the task load and the system usability score (SUS) [6] to measure the usability of the methods. Each participant filled out the questionnaires after they completed the task.

5.5 Results

5.5.1 Objective Metrics

Fig. 10 gives the results for each objective metric. We conducted a multivariate analysis of variance (MANOVA) to analyze the effects of three independent variables (group selection method, number of groups m, and the layout orderedness $\mathscr{S}$) on the performance of the above metrics. For each trial condition, the outlier data points were first filtered out (± 3 standard deviation). We removed 13 data points in total (2.67%). The outlier numbers for each condition are marked at the bottom of the bars of the completion time. Before analysis, Shaipro-Wilk's test showed that the two dependent variables (completion time and number of operations) obeyed a normal distribution ($p > 0.05$).

Interaction Effect Overall, the interaction effect of the three independent variables was statistically significant ($F_{32,1683} = 26.942, p < 0.001, \Lambda_{Wilks} = 0.218, \eta_p^2 = 0.317$). Particularly, there was a statistically significant effect of the interaction of group selection method and group number on the dependent variables ($F_{16,1394} = 12.71, p < 0.001, \Lambda_{Wilks} = 0.66, \eta_p^2 = 0.099$). The interaction of the group selection method and group number on completion time ($F_{4,459} = 20.166, p < 0.001, \eta_p^2 = 0.149$) and number of operations ($F_{4,459} = 14.905, p < 0.001, \eta_p^2 = 0.115$) were all statistically significant.

The interaction effect of the group selection method and layout orderedness on the dependent variables was also statistically significant ($F_{16,1394} = 35.371, p < 0.001, \Lambda_{Wilks} = 0.353, \eta_p^2 = 0.229$). The interaction of the group selection method and layout orderedness on completion time ($F_{4,459} = 9.65, p < 0.001, \eta_p^2 = 0.074$) and number of operations ($F_{4,459} = 15.686, p < 0.001, \eta_p^2 = 0.120$) were all statistically significant.

The interaction effect of the group number and layout orderedness on the dependent variables was also statistically significant ($F_{16,1394} = 37.458, p < 0.001, \Lambda_{Wilks} = 0.335, \eta_p^2 = 0.239$). The interaction of the group number and layout orderedness on completion time ($F_{4,459} = 32.251, p < 0.001, \eta_p^2 = 0.219$) and number of operations ($F_{4,459} = 21.124, p < 0.001, \eta_p^2 = 0.155$) were all statistically significant.

Main Effect Multivariate tests show that the main effect of the group selection method on the dependent variables was statistically significant ($F_{8,912} = 1505.116, p < 0.001, \Lambda_{Wilks} = 0.005, \eta_p^2 = 0.930$). Univariate main effect tests revealed statistically significant effects of the group selection method on the completion time

This article has been accepted for publication in IEEE Transactions on Visualization and Computer Graphics. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TVCG.2025.3549560

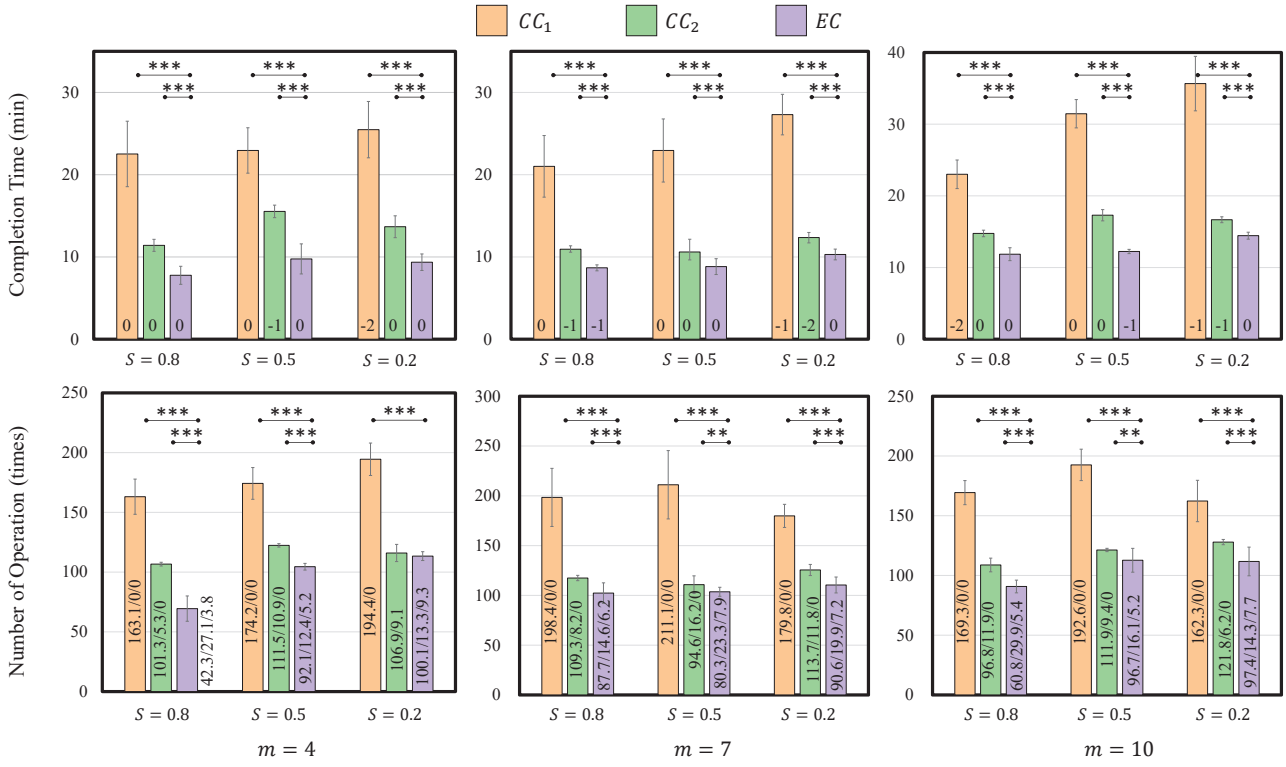


Fig. 10: Objective metrics results. The top row shows the completion time, and the bottom row shows the number of operations of different m and $\mathscr{S}$ layouts for three comparing methods. Different colors distinguish the methods. The error line demonstrates the standard deviation. The outlier numbers for each condition are marked at the bottom of the bars of the completion time. Numbers of single-/multi-selection/PwP-round times are marked inside the bars of the bottom row. Significant differences are denoted with asterisks (all compared to the EC method).

($F_{2,459} = 1923.741, p < 0.001, \eta_p^2 = 0.893$) and the number of operations ($F_{2,459} = 1840.177, p < 0.001, \eta_p^2 = 0.889$). Pairwise comparison showed statistically significant differences ($p < 0.001$) among the three group selection methods on all two objective metrics.

The main effect of the group number on the dependent variables was also statistically significant ($F_{8,912} = 38.059, p < 0.001, \Lambda_{Wilks} = 0.562, \eta_p^2 = 0.250$). Univariate main effect tests revealed statistically significant effects of the group number on the completion time ($F_{2,459} = 99.836, p < 0.001, \eta_p^2 = 0.303$) and the number of operations ($F_{2,459} = 28.864, p < 0.001, \eta_p^2 = 0.112$). Pairwise comparison showed statistically significant differences among the three group numbers on completion time ($p < 0.021$) and number of operations ($p < 0.025$).

The main effect of the layout orderedness on the dependent variables was also statistically significant ($F_{8,912} = 49.370, p < 0.001, \Lambda_{Wilks} = 0.487, \eta_p^2 = 0.302$). Univariate main effect tests revealed statistically significant effects of the layout orderedness on the completion time ($F_{2,459} = 52.486, p < 0.001, \eta_p^2 = 0.186$) and the number of operations ($F_{2,459} = 60.946, p < 0.001, \eta_p^2 = 0.210$). Pairwise comparison showed no statistically significant differences between $\mathscr{S} = 0.2$ and $\mathscr{S} = 0.5$ on completion time ($p = 0.231$) and number of operations ($p = 0.632$). There were statistically significant differences between $\mathscr{S} = 0.2$ and $\mathscr{S} = 0.8$, and $\mathscr{S} = 0.5$ and $\mathscr{S} = 0.8$ on the two dependent variables ($p < 0.001$).

Comparison of EC For the completion time metric, the effect of the group number shows there were significant ($p < 0.001$) differences between $m = 4$ and $m = 10$, and $m = 7$ and $m = 10$, except for $m = 4$ and $m = 7$ ($p = 1.0$). The effect of the layout orderedness shows there were significant differences between $\mathscr{S} = 0.2$ and $\mathscr{S} = 0.5$ ($p = 0.018$), and $\mathscr{S} = 0.2$ and $\mathscr{S} = 0.8$ ($p < 0.001$), except for $\mathscr{S} = 0.5$ and $\mathscr{S} = 0.8$ ($p = 0.1$).

For the number of operation metrics, the effect of the group number shows there were significant ($p < 0.001$) differences between $m = 4$ and $m = 7$, and $m = 4$ and $m = 10$, except for $m = 7$ and $m = 10$ ($p = 1.0$). The effect of the layout orderedness shows there were significant differences between $\mathscr{S} = 0.2$ and $\mathscr{S} = 0.8$ ($p < 0.001$), and $\mathscr{S} = 0.5$ and $\mathscr{S} = 0.8$ ($p < 0.001$), except for $\mathscr{S} = 0.2$ and $\mathscr{S} = 0.5$ ($p = 0.137$).

5.5.2 Subjective Metrics

In Fig. 11, we report the NASA-TLX scores and the SUS scores for the three methods. The Holm-Bonferroni corrected post hoc pairwise t-test is conducted to compare the subscores between the CC methods and the EC method. For the overall NASA-TLX score and the SUS score, the EC method shows significant improvement ($p < 0.01$) compared to the CC methods. Regarding the subscores of NASA-TLX, the EC method needs more *Mental Demand*, and the differences are significant ($p < 0.05$) compared to the CC methods. Moreover, for *Effort* and *Frustration* subscores, there are no significant differences ($p = 0.78$, $p = 0.061$) between the EC and CC_2 methods, while the differences are significant ($p < 0.001$) between the EC and CC_1 methods. For the remaining subscores, the EC method shows statistically significant improvement.

5.6 Discussion

In conclusion, our proposed PwP method efficiently selects groups in environments with complex object layouts. During operation, probability-based grouping effectively reduces the number of user operations, accelerating the group selection process and reducing the user workload, exhibiting better system availability.

Regarding group selection efficiency improvement, our approach is mainly realized by increasing the number of batch selections in the group selection process through probabilistic propagation and object rearrangement clustering. From the results of the single-/batch-selection/PwP-module times, we can see that the EC method is able to make more batch selections during the task compared to the CC

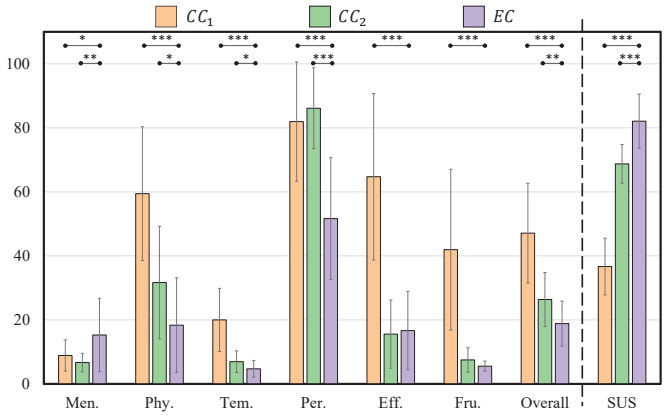


Fig. 11: Plots for NASA-TLX subscores (left of the vertical dash line) and SUS scores (rightmost column) of the compared three methods. Statistically significant differences are marked with asterisks.

methods. For a scene with high complexity, the level of orderedness of the objects increases with each round of the PwP module, enabling the user to more easily employ batch selection for object group selection. This significantly differs from the CC_2 method, which does not change the object groups' orderedness throughout the process. Compared to the CC_1 method, MultiFingerBubble requires users to use multiple fingers to select an object each time, which puts a heavy burden on users. Although the accuracy of the operation can be improved by optimizing the objects referred to between fingers, many wrong selections were still made. We believe these are the reasons why the PwP method can stand out in the comparison of group selection efficiency. The results support **H1**.

In addition, for the PwP method, there was no significant change in completion time when $m = 4$ and $m = 7$, and there was a more significant increase in completion time when m increased to 10. Because the more groups there are, the more difficult it is to conduct batch selection. As the orderedness decreases, i.e., the more scattered the objects in the scene, the longer the completion time becomes. When orderedness is greater than 0.5, the increase in completion time levels off. This is because the PwP method's time consumption is concentrated in the first selection round. When orderedness is larger than 0.5, there are fewer batch selections in this round, and the increase in time consumption is mainly related to the total number of objects n and the group number m.

For the number of operations, compared to the CC_2 method, our method exhibits a smaller number of operations due to the fact that more batch selections were performed. In terms of the CC_1 method, due to the difficulty of multi-finger manipulation, users often made incorrect selections, and the indicator of the number of operations was larger due to the presence of multiple modifications. The results support **H2**.

In terms of task load, our method does not dominate in *Mental Demand* compared to the other two methods. We think this may be related to the fact that users are looking for better areas to perform batch selection during the selection process, and they hope to find more objects of the same group clustered together. This may be the reason why their MD subscore is relatively higher. For the rest of the metrics, our method shows better task load reduction. Because we have increased the number of convenient batch selections, we have significantly reduced the time to complete the task and demonstrated more advantageous user task load performance. The above enhancements result in better user usability of our approach.

6 Conclusion, Limitations and Future works

We have proposed a novel permutating with probability method aiming at efficient group selection tasks in an immersive virtual environment. Our PwP method introduces probability propagation and object position rearrangement algorithms to update the object layout over multiple rounds for fast user selection. We designed ablation experiments to determine the coefficients involved in the algorithms. We carried out a user study to verify that our method significantly improves the efficiency of the group selection task compared to traditional multi-object selection methods while reducing the user task load and exhibiting better usability. Our PwP method enables fast object selection and grouping in virtual environments with complex layouts. The grouping enables efficiency improvement in subsequent virtual reality interaction tasks such as multi-object selection, manipulation, alignment, and arrangement.

The grouping in our method is not based on predefined object features, such as injured soldiers or metal objects, but rather on the features that emerge through user interactions. Our PwP module relies on the spatial position of the object selected by the user, along with the position information of surrounding objects, their spatial relationships, and the implicit probabilistic features established by the user during the interaction. This approach provides flexibility, particularly for creative applications where users (such as artists, architects, etc.) can adjust the grouping method to suit their own preferences. For instance, users can freely group cubes and cats together as one category, and dogs and spheres as another. The only requirement is that the objects they wish to group are marked with the same color during the selection process.

One limitation of the method is that the current algorithm does not consider the case where the user is inside the total object bounding box. Since each round of object position updating is related to the user's selection, if the user is inside the enclosing box, there will be a situation where some objects fall in the user's viewpoint or behind the user when calculating the updated object position, resulting in a difficult selection in the next round. Another limitation is that for real-time considerations, in the process of probability propagation, we calculate all the objects in front of the target object in the queue without considering the spatial distance between the two objects and the topological relationship in the virtual environment. Another limitation is that our current solution involves gathering objects into corners and edges to expand the distance between each group of objects. However, this layout may cause an occlusion problem. When unselected objects are rearranged between the front and back groups, the front group can block them, making selection more difficult.

Future work could explore how to adjust the object position rearrangement algorithm when the user is inside the total bounding box or assign the spatial grid occupied by the user as already rearranged objects, which would help prevent occlusion.. Another future work could describe the spatial relationships of objects in the virtual environment by building a graph and then considering the distances and connections between objects when calculating their new positions. For the occlusion problem, an effective solution is to adopt a dispersed layout within the user's field of view, which can help minimize the chances of objects being obstructed.

Acknowledgments

This work was supported by the National Natural Science Foundation of China through Projects 62372026 and 61932003, by Beijing Science and Technology Plan Project Z221100007722004, and by NationalKey R&D plan 2019YFC1521102.

References

[1] F. Argelaguet and C. Andujar. A survey of 3d object selection techniques for virtual environments. *Computers & Graphics*, 37(3):121–136, 2013. doi: 10.1016/j.cag.2012.12.003 2

[2] M. Baloup, T. Pietrzak, and G. Casiez. Raycursor: A 3d pointing facilitation technique based on raycasting. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, CHI '19, 12 pages, p. 1–12. Association for Computing Machinery, New York, NY, USA, 2019. doi: 10.1145/3290605.3300331 2

[3] Y. Bao, J. Wang, Z. Wang, and F. Lu. Exploring 3d interaction with gaze guidance in augmented reality. In *2023 IEEE Conference Virtual Reality and 3D User Interfaces (VR)*, pp. 22–32. IEEE, 2023. 2

[4] P. Boffi, A. Kouyoumdjian, M. Waldner, P. L. Lanzi, and I. Viola. Bagginhook: Selecting moving targets by pruning distractors away for

intention-prediction heuristics in dense 3d environments. In *2024 IEEE Conference Virtual Reality and 3D User Interfaces (VR)*, pp. 913–923. IEEE, 2024. 2

[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*, pp. 35–ff, 1997. 1, 2

[6] J. Brooke. Sus: A "quick and dirty" usability scale. *Usability Evaluation in INdustry/Taylor and Francis*, 1996. 7

[7] 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, 2012. 2

[8] D. L. Chen, M. Giordano, H. Benko, T. Grossman, and S. Santosa. Gazeraycursor: Facilitating virtual reality target selection by blending gaze and controller raycasting. In *Proceedings of the 29th ACM Symposium on Virtual Reality Software and Technology*, pp. 1–11, 2023. 2

[9] K. Chen, H. Wan, S. Zhao, and X. Liu. Backtracer: Improving ray-casting 3d target acquisition by backtracking the interaction history. *International Journal of Human-Computer Studies*, 176:103045, 2023. 2

[10] X. Chen, A. Acharya, A. Oulasvirta, and A. Howes. An adaptive model of gaze-based selection. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, pp. 1–11, 2021. 2

[11] M. Ciolfi Felice, N. Maudet, W. E. Mackay, and M. Beaudouin-Lafon. Beyond snapping: Persistent, tweakable alignment and distribution with stickylines. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*, pp. 133–144, 2016. 2

[12] G. De Haan, M. Koutek, and F. H. Post. Intenselect: Using dynamic object rating for assisting 3d object selection. In *Ipt/egve*, pp. 201–209, 2005. 2

[13] H. Debarba, L. Nedel, and A. Maciel. Lop-cursor: Fast and precise interaction with tiled displays using one hand and levels of precision. In *2012 IEEE Symposium on 3D User Interfaces (3DUI)*, pp. 125–132, 2012. doi: 10.1109/3DUI.2012.6184196 2

[14] W. Delamare, M. Daniel, and K. Hasan. Multifingerbubble: A 3d bubble cursor variation for dense environments. In *CHI Conference on Human Factors in Computing Systems Extended Abstracts*, pp. 1–6, 2022. 1, 2, 7

[15] A. O. S. Feiner. The flexible pointer: An interaction technique for selection in augmented and virtual reality. In *Proc. UIST*, vol. 3, pp. 81–82, 2003. 2

[16] S. G. Hart. Nasa-task load index (nasa-tlx); 20 years later. In *Proceedings of the human factors and ergonomics society annual meeting*, vol. 50, pp. 904–908. Sage publications Sage CA: Los Angeles, CA, 2006. 7

[17] S. G. Hart and L. E. Staveland. Development of nasa-tlx (task load index): Results of empirical and theoretical research. In *Advances in psychology*, vol. 52, pp. 139–183. Elsevier, 1988. 7

[18] 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. IEEE, 2011. 2

[19] Y. Li, S. Sarcar, K. Kim, H. Tu, and X. Ren. Designing successive target selection in virtual reality via penetrating the intangible interface with handheld controllers. *International Journal of Human-Computer Studies*, 165:102835, 2022. 2

[20] T. Luong, Y. F. Cheng, M. Möbus, A. Fender, and C. Holz. Controllers or bare hands? a controlled evaluation of input techniques on interaction performance and exertion in virtual reality. *IEEE Transactions on Visualization and Computer Graphics*, 2023. 2

[21] M. Maslych, Y. Hmaiti, R. Ghamandi, P. Leber, R. K. Kattoju, J. Belga, and J. J. LaViola. Toward intuitive acquisition of occluded vr objects through an interactive disocclusion mini-map. In *2023 IEEE Conference Virtual Reality and 3D User Interfaces (VR)*, pp. 460–470. IEEE, 2023. 2

[22] D. Mendes, F. M. Caputo, A. Giachetti, A. Ferreira, and J. Jorge. A survey on 3d virtual object manipulation: From the desktop to immersive virtual environments. In *Computer graphics forum*, vol. 38, pp. 21–45. Wiley Online Library, 2019. 2

[23] R. A. Montano-Murillo, C. Nguyen, R. H. Kazi, S. Subramanian, S. DiVerdi, and D. Martinez-Plasencia. Slicing-volume: Hybrid 3d/2d multitarget selection technique for dense virtual environments. In *2020 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 53–62. IEEE, 2020. 2

[24] A. K. Mutasim, A. U. Batmaz, and W. Stuerzlinger. Pinch, click, or dwell: Comparing different selection techniques for eye-gaze-based pointing in virtual reality. In *Acm symposium on eye tracking research and applications*, pp. 1–7, 2021. 2

[25] J.-Y. Oh, W. Stuerzlinger, and D. Dadgari. Group selection techniques for efficient 3d modeling. In *3D User Interfaces (3DUI'06)*, pp. 95–102. IEEE, 2006. 2

[26] J. S. Pierce, B. C. Stearns, and R. Pausch. Voodoo dolls: seamless interaction at multiple scales in virtual environments. In *Proceedings of the 1999 symposium on Interactive 3D graphics*, pp. 141–145, 1999. 2

[27] H. Pohl, K. Lilija, J. McIntosh, and K. Hornbæk. Poros: configurable proxies for distant interactions in vr. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, pp. 1–12, 2021. 2

[28] 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*, pp. 79–80, 1996. 2

[29] M. Sereno, S. Gosset, L. Besançon, and T. Isenberg. Hybrid touch/tangible spatial selection in augmented reality. In *Computer Graphics Forum*, vol. 41, pp. 403–415. Wiley Online Library, 2022. 2

[30] R. Shi, J. Zhang, W. Stuerzlinger, and H.-N. Liang. Group-based object alignment in virtual reality environments. In *Proceedings of the 2022 ACM Symposium on Spatial User Interaction*, pp. 1–11, 2022. 2

[31] R. Shi, J. Zhang, Y. Yue, L. Yu, and H.-N. Liang. Exploration of bare-hand mid-air pointing selection techniques for dense virtual reality environments. In *Extended Abstracts of the 2023 CHI Conference on Human Factors in Computing Systems*, pp. 1–7, 2023. 2

[32] L. Sidenmark, C. Clarke, J. Newn, M. N. Lystbæk, K. Pfeuffer, and H. Gellersen. Vergence matching: Inferring attention to objects in 3d environments for gaze-assisted selection. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, pp. 1–15, 2023. 2

[33] L. Sidenmark, C. Clarke, X. Zhang, J. Phu, and H. Gellersen. Outline pursuits: Gaze-assisted selection of occluded objects in virtual reality. In *Proceedings of the 2020 chi conference on human factors in computing systems*, pp. 1–13, 2020. 2

[34] L. Sidenmark, M. Parent, C.-H. Wu, J. Chan, M. Glueck, D. Wigdor, T. Grossman, and M. Giordano. Weighted pointer: Error-aware gaze-based interaction through fallback modalities. *IEEE Transactions on Visualization and Computer Graphics*, 28(11):3585–3595, 2022. 2

[35] L. Sidenmark, F. Prummer, J. Newn, and H. Gellersen. Comparing gaze, head and controller selection of dynamically revealed targets in head-mounted displays. *IEEE Transactions on Visualization and Computer Graphics*, 2023. 2

[36] R. Stenholt. Efficient selection of multiple objects on a large scale. In *Proceedings of the 18th ACM symposium on Virtual reality software and technology*, pp. 105–112, 2012. 2

[37] 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*, pp. 265–272, 1995. 2

[38] W. Stuerzlinger and G. Smith. Efficient manipulation of object groups in virtual environments. In *Proceedings IEEE Virtual Reality 2002*, pp. 251–258. IEEE, 2002. 2

[39] 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, 2007. 2

[40] U. Wagner, M. N. Lystbæk, P. Manakhov, J. E. S. Grønbæk, K. Pfeuffer, and H. Gellersen. A fitts' law study of gaze-hand alignment for selection in 3d user interfaces. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, pp. 1–15, 2023. 2

[41] 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. IEEE, 2021. 2

[42] Y. Wei, R. Shi, D. Yu, Y. Wang, Y. Li, L. Yu, and H.-N. Liang. Predicting gaze-based target selection in augmented reality headsets based on eye and head endpoint distributions. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, pp. 1–14, 2023. 2

[43] H. Wu, X. Sun, H. Tu, and X. Zhang. Clockray: A wrist-rotation based technique for occluded-target selection in virtual reality. *IEEE Transactions on Visualization and Computer Graphics*, 2023. 1, 2

[44] J. Wu, L. Wang, S. K. Im, and C. T. Lam. Eeba: Efficient and ergonomic big-arm for distant object manipulation in vr, 2024. 2

[45] Z. Wu, D. Yu, and J. Goncalves. Point-and volume-based multi-object acquisition in vr. In *IFIP Conference on Human-Computer Interaction*, pp. 20–42. Springer, 2023. 1, 2, 7

[46] H. P. Wyss, R. Blach, and M. Bues. isith-intersection-based spatial interaction for two hands. In *3D User Interfaces (3DUI'06)*, pp. 59–61. IEEE, 2006. 2

[47] 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, 2020. doi: 10.1109/TVCG.2020.3023606 2

[48] L. Yu, K. Efstathiou, P. Isenberg, and T. Isenberg. Cast: Effective and efficient user interaction for context-aware selection in 3d particle clouds. *IEEE transactions on visualization and computer graphics*, 22(1):886–895, 2015. 2

[49] Q. Zhang, C.-H. Kim, and H. W. Byun. Multi-finger-based arbitrary region-of-interest selection in virtual reality. *International Journal of Human–Computer Interaction*, 39(20):3969–3983, 2023. 2

[50] L. Zhao, T. Isenberg, F. Xie, H.-N. Liang, and L. Yu. Metacast: Target-and context-aware spatial selection in vr. *IEEE Transactions on Visualization and Computer Graphics*, 2023. 2

[51] F. Zhu, L. Sidenmark, M. Sousa, and T. Grossman. Pinchlens: Applying spatial magnification and adaptive control-display gain for precise selection in virtual reality. In *2023 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pp. 1221–1230. IEEE, 2023. 2