MOA: Efficient Scene-aware Multi-object Arrangement in VR

Xuehuai Shi, Yuhan Duan, Ziteng Wang, Jian Wu, Zhiwen Shao, Jieming Yin, and Lili Wang

***Abstract*—3D multi-object arrangement is a fundamental task in VR that relies on accurate and natural initial selection alongside rapid and convenient subsequent manipulation to ensure high efficiency. However, existing methods fail to support efficient multi-object arrangement in highly occluded scenes with densely packed candidate objects through controller-free natural interactions. In this paper, we propose an efficient, scene-aware multi-object arrangement method (MOA) designed for fast, precise, and convenient object arrangement. First, MOA introduces an importance-driven multi-object initial selection algorithm that assigns higher spatiotemporally correlated object importance (IMP) to target objects, establishing a natural multi-object initial selection mode that enables quick and accurate selection of high-IMP objects. Then, it presents an auxiliary-structure-guided multi-object manipulation algorithm that constructs an auxiliary manipulation structure to assist subsequent multi-object manipulation, alongside a multi-modal interaction mode that facilitates swift and natural manipulation. Compared to state-of-the-art methods, MOA significantly improves task performance, reduces task load, and enhances convenience in complex multi-object arrangement scenes involving hundreds of highly occluded objects that need to be arranged.**

***Index Terms*—Virtual Reality, Multi-object Arrangement, Controller-free Interaction, VR Interaction.**

I. Introduction

In virtual reality (VR), 3D multi-object arrangement is a fundamental task widely applied in various fields such as educational training [1]–[3], 3D design [4]–[6], and game development [7]–[9]. Currently, the growing demand for natural interaction in VR is making controller-free interaction mainstream. As VR scenes become more complex and applications diversify, developing novel controller-free interaction techniques that enable convenient, quick, and accurate *selection* and *manipulation* of multiple target objects is key to achieving efficient 3D multi-object arrangement. For example, in a virtual museum exhibit layout, conveniently and accurately selecting all displayed cultural relics from the warehouse through natural interaction and quickly placing them at designated target positions within the exhibition hall requires an efficient multi-object arrangement technique.

Interaction in multi-object arrangement consists of two primary steps: *initial selection* and *later manipulation* of target objects [10]–[12]. However, existing multi-object arrangement techniques are not efficient enough in complex VR scenes with high occlusion and dense candidate objects. This inefficiency means that these methods' precision, performance, task load, and convenience fail to meet users' needs for a natural, immersive user experience in VR. Current *initial selection* methods mainly rely on concepts such as the bubble cursor [13]–[15] and gaze-assisted interaction [12], [16], [17] to facilitate object selection in VR scenes. However, when selecting highly occluded target objects in dense regions, these methods often mistakenly select non-target objects due to limitations in finger and visual sensitivity and thus cannot quickly select multiple target objects. Existing *later manipulation* approaches focus on placing target objects by leveraging physical-virtual motion alignment [17]–[19] and multiple manipulation points [20]–[22]. However, these methods lack guidance and natural interaction modes during later manipulation, leaving room for improvement in both performance and convenience.

To achieve efficient multi-object arrangement in complex scenes with high occlusion and dense objects using controller-free natural interaction modes, two challenges need to be addressed. The first challenge is to leverage user attention to improve the accuracy of selecting multiple target objects and to develop a controller-free selection mode that enhances this selection performance. The second challenge is to utilize the spatial correlations of target positions to guide the manipulation of multiple target objects and to create a convenient controller-free manipulation mode that accelerates their manipulation.

In this paper, we propose an efficient scene-aware multi-object arrangement method (MOA) to rapidly and conveniently select a large number of target objects and manipulate them to target positions in complex scenes. To address the first challenge, we propose the importance-driven multi-object initial selection algorithm (MOA_s). MOA_s calculates the spatiotemporally correlated object importance (IMP) of all objects based on user attention, enhancing the target objects' IMP. Then, MOA_s constructs a natural multi-object initial selection mode that achieves rapid and precise selection of objects with high importance. To address the second challenge, we propose the auxiliary-structure-guided multi-object later manipulation algorithm MOA_m. MOA_m constructs an auxiliary structure to predict the potential positions of the remaining target objects based on the known target positions. Then, MOA_m creates a

Xuehuai Shi is with the State Key Laboratory of Tibetan Intelligent Information Processing and Application, School of Artificial Intelligence, Nanjing University of Posts and Telecommunications, Jiangsu, China, 210023.

Yuhan Duan, Ziteng Wang, Jian Wu and Lili Wang are with the State Key Laboratory of Virtual Reality Technology and Systems, School of Computer Science and Engineering, Beihang University, Beijing, 100191, China, and also with Peng Cheng Laboratory, Shenzhen, Guangdong, 518000, China.

Zhiwen Shao is with the School of Computer Science and Technology, China University of Mining and Technology, Jiangsu, 221116, China.

Jieming Yin is with the School of Computer Science, Nanjing University of Posts and Telecommunications, Jiangsu, 210023, China.

Ziteng Wang and Jian Wu are the corresponding authors. E-mail: 21373237@buaa.edu.cn, lanayawj@buaa.edu.cn.

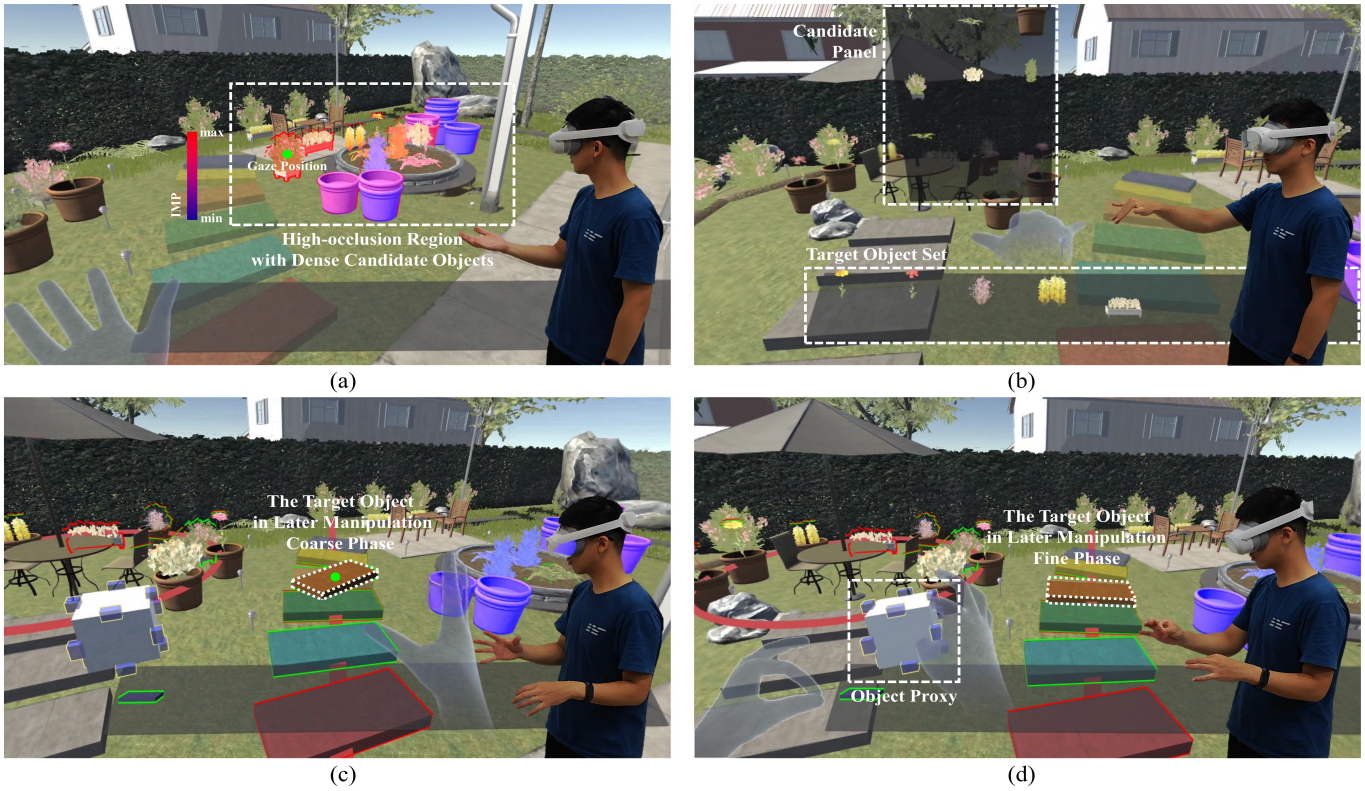


Fig. 1. A user employs the efficient scene-aware multi-object arrangement method (MOA) for multi-object arrangement in *garden*. (a) Firstly, after MOA calculates the spatiotemporally correlated object importance (IMP) for all candidate objects based on user attention, the user enters the natural multi-object initial selection mode using a right-hand 'lifting' gesture. (b) Secondly, the user quickly and accurately selects all target objects by leveraging the natural multi-object initial selection mode based on IMP. (c) Thirdly, guided by the auxiliary structure (marked as the red circle and line), the user achieves rapid coarse manipulation of target objects using the multi-modal multi-object later manipulation mode's coarse phase. (d) Finally, the user conveniently manipulates the target objects to their corresponding target positions using the multi-modal multi-object later manipulation mode's fine phase.

multi-modal multi-object later manipulation mode that quickly guides target objects to their corresponding positions based on the auxiliary structure. Fig. 1 illustrates the process of a user performing multi-object arrangement using MOA.

In summary, the contributions of this paper are as follows:

- We propose the MOA$_s$, which models object importance to prioritize the presentation of objects most likely to be targets and creates a controller-free selection mode that enables efficient multi-object initial selection.
- We propose the MOA$_m$, which constructs an auxiliary structure to guide object manipulation and establishes a controller-free manipulation mode to accelerate multi-object later manipulation.
- We conduct several user studies to evaluate the accuracy, performance, and user experience of the proposed method.

Source code is available online [1].

II. Related Work

MOA draws on a variety areas of prior research, particularly multi-object initial selection techniques, multi-object later manipulation techniques, and controller-free interaction techniques in VR. We elaborate on the recent related work of these three techniques in this section.

A. Multi-object Initial Selection

Multi-object initial selection is a fundamental interaction task in VR. However, directly selecting target objects is inefficient in complex scenes where target objects are occluded. Researchers propose various techniques to improve selection efficiency and accuracy by managing complexity and mitigating occlusion during initial selection.

Wang et al. [23] introduce a selection method based on eliminating occlusions in the user's central field of view to enhance quick target object selection in high-occlusion scenes with dense target objects. Wu et al. [24] first introduce a multi-perspective visualization method to reduce user movement distance during multi-target object selection. To further enhance selection performance, they remove fine-grained occlusions to achieve quick target object selection in regions with dense objects [25]. Bhowmick et al. [26] propose the tiny hands technique to efficiently navigate and accurately select the small target objects in dense VR scenes. Zhu et al. [27] propose a new freehand target selection method to enhance user performance in selecting target objects among numerous small, dense options. Jiang et al. [28] propose a knowledge-driven joint reasoning approach for object selection. Chen et al. [29] improve the ray-tracing-based object selection technique by tracing the interaction history back to the time when the target object was highlighted when selecting objects using ray tracing. To enable the simultaneous selection of multiple target objects, Delamare et al. [13] propose a multi-finger 3D bubble

[1]https://drive.google.com/file/d/1nbZCqMX9OQyJHIn8SrNNFzXeN-EDRces/view?usp=drive_link

cursor technique to select multiple target objects in the scene concurrently by projecting several rays from specific fingers.

Several typical techniques and strategies are widely used in VR for multi-object initial selection, such as World-In-Miniature (WIM) [30], Cone [31], and the progressive refinement strategy [32]. Stoakley et al. [30] first introduce the milestone method WIM, which provides users with a valuable overview by constructing a miniature version of the current virtual environment, allowing proxy interactions with distant objects. However, when the virtual environment contains highly overlapping selectable objects, users find it difficult to accurately select target objects in the miniature world. Maslych et al. [31] improve WIM by utilizing conical projection to reduce the presentation range of the miniature world and propose a scale function to arrange objects within the conical range for selection. However, this approach assumes that the probability of each selectable object being a target is uniform, overlooking inconsistencies in target object probabilities. This limitation makes it difficult to facilitate selection of the most likely target objects, leading to inefficiencies in large-scale multi-object initial selection. Kopper et al. [32] introduce a progressive refinement strategy that breaks selections into multiple stages using explicit criteria such as spatial volume or menus, ultimately aiming to select target objects. However, because users' decisions at each stage produce different partitioning outcomes, they must actively decide the direction of each partition. In high-occlusion environments requiring many decisions, this increases task difficulty and cognitive load.

We propose MOA$_s$, which leverages implicit cues from user attention to identify the most probable target objects in complex scenes containing hundreds of candidates and significant occlusion. It prioritizes the presentation of the most likely targets and enables users to quickly and accurately select them through a convenient, controller-free interaction mode. Compared to WIM, MOA$_s$ avoids the cumbersome process of requiring users to carefully select target objects one by one among highly overlapping items. Compared to Cone, MOA$_s$ swiftly highlights the most likely target objects, simplifying the selection process. Compared to the progressive refinement strategy, MOA$_s$ intelligently achieves precise segmentation of target objects based on users' implicit gaze motion, eliminating the need for users to actively decide segmentation direction, thereby effectively reducing the cognitive load associated with user-driven decision sequences.

B. Multi-object Later Manipulation

Recent studies improve existing 3D manipulation methods for tasks involving multiple target objects in VR, aiming to enhance efficiency and user experience when manipulating many targets in complex VR scenes.

Wang et al. [33] introduce a collaborative manipulation method that enhances performance by providing better perspectives during the manipulation of target objects, thereby improving multi-object manipulation tasks. Li et al. [34] achieve multi-object manipulation for robots using relational reinforcement learning, performing well in simultaneously manipulating blocks into a tower. Shi et al. [35] propose several group-based 3DoF translation alignment interaction techniques in VR, enabling rapid translation and precise alignment of objects within groups. Li et al. [36] present a swarm manipulation method based on swarm control theory, allowing users to manipulate multiple virtual objects by controlling a group of particles with right-hand gestures. Wu et al. [37] propose an efficient and ergonomic big-arm method, which extends the upper arm and forearm based on the maximum operational space distance. To optimize the performance, accuracy, and comfort of multi-object manipulation, Zheng et al. [17] propose an object manipulation method in VR based on variable virtual interaction regions, named VVIR. Compared with state-of-the-art methods, VVIR significantly improves completion time and manipulation precision while reducing fatigue during multi-object manipulation tasks.

State-of-the-art methods for multi-object manipulation overlook the spatial correlations between target positions, failing to leverage this information to enhance multi-object manipulation performance for remaining target objects. Controller-free natural manipulation modes are also lacking in terms of multi-object manipulation. To address these issues, we propose MOA$_m$, which constructs and dynamically updates an auxiliary structure based on the known target positions to guide the manipulation of the remaining target objects. Additionally, we introduce a multi-modal multi-object later manipulation mode, enabling natural multi-object manipulation without controllers, thereby further improving the accuracy and convenience of multi-object manipulation.

C. Controller-free Interaction in VR

Controller-free interaction in VR refers to using techniques such as gesture recognition, gaze tracking, and motion capture to directly interact with virtual content, instead of relying on traditional physical controllers or joysticks for manipulating 3D objects. Recent research in this area is mainly divided into two categories. The first category explores new interaction techniques that enhance the user experience by combining novel multi-modal inputs such as gestures, gaze, and voice. The second category focuses on optimizing object interaction algorithms to improve the precision of grasping, translating, rotating, and scaling objects, thereby enhancing interaction performance and the user's sense of control.

In the recent research on new interaction technology exploration, Song et al. [38] are the first to implement object interaction in VR using bare-hand gestures, which utilizes the relative 3D motion of both hands to enable rapid manipulation and alignment of single objects. Arora et al. [39] create and edit dynamic physical phenomena in VR, such as particle systems, deformations, and coupling. Yu et al. [12] develop a gaze-assisted strategy for bare-hand manipulation of 3D objects to enhance integration, coordination, and transition in bare-hand manipulation. Dewez et al. [40] propose practical guidelines to improve the user's sense of embodiment. Bozgeyikli et al. [41] introduce a time-multiplexed tangible user interface for manipulating virtual objects. Wu et al. [24], [25] propose a head pose-driven fine-grained occlusion removal method to facilitate quick target object identification. Lee et al. [42]

build a gesture-driven system for creating and modifying 3D curve networks. Luong et al. [43] evaluate the performance of controller-based and bare-hand manipulation methods in two mid-air interaction techniques (touch and raycast).

In recent work on optimizing object interaction algorithms, Chen et al. [44] propose a bare-hand interaction algorithm based on multi-modal input to eliminate ambiguities caused by imprecise gesture tracking. Moran-Ledesma et al. [45] use context-free grammar to eliminate ambiguities in prop-based gesture descriptions. Pei et al. [46] introduce Hand Interfaces to quickly create virtual objects in VR scenes. Zhang et al. [47] improve bare-hand manipulation performance and user experience in grasping tasks by adjusting audio in VR space. Meng et al. [48] explore hands-free text selection interaction in VR HMDs. Yu et al. [49] design six manipulation modes combining body and air interfaces for more efficient 3D object scaling, rotation, and movement. Lee et al. [50] introduce a gaze-based text reading system to improve text reading performance in VR.

Controller-free interaction techniques gradually become mainstream natural interaction methods in VR. However, during the multi-object arrangement in VR, existing technologies struggle to quickly and accurately select numerous target objects from a large pool. Therefore, the proposed natural multi-object initial selection mode and multi-modal multi-object later manipulation mode in the proposed MOA provides a controller-free interaction scheme that enables efficient multi-object arrangement in complex VR scenes.

III. Efficient Scene-aware Multi-object Arrangement in VR

In this section, we propose the MOA to achieve a rapid and convenient multi-object arrangement in complex VR scenes. We introduce the importance-driven multi-object initial selection algorithm MOA$_s$ in Section III-A, and present the auxiliary-structure-guided multi-object later manipulation algorithm MOA$_m$ in Section III-B. The interaction mode designs in both MOA$_s$ and MOA$_m$ fully consider users' interaction habits, ensuring intuitive and effortless operation. We select four commonly used gestures from daily life (thumb up, lift, click, and pinch) to establish a comprehensive multi-object arrangement interaction mode that satisfies the requirements for natural interaction and immersive user experience. All subsequent studies in this research are approved by the Biology and Medical Ethics Committee of Beihang University.

A. Importance-driven Multi-object Initial Selection

We propose the MOA$_s$ to conveniently accelerate multi-object selection in high-occlusion scenes with densely packed objects. MOA$_s$ models the spatiotemporally correlated object importance of all objects based on user attention, and introduces a natural multi-object initial selection mode to efficiently accelerate multi-object selection based on IMP.

Algorithm 1 details the workflow of MOA$_s$. Given a 3D scene S, the user's current viewpoint V and gaze position $gaze$, the right-hand gesture $gestR$, the left-hand gesture $gestL$, the predefined maximum number of candidate objects $\#N_c$, and the importance coefficient α, MOA$_s$ outputs the selected target object set OBJ. MOA$_s$ consists of two steps: the spatiotemporally correlated object importance (IMP) calculation (lines 1-10), which is detailed in Section III-A1; and the target object selection via a natural selection mode (lines 11-16), which is detailed in Section III-A2.

Algorithm 1 Importance-driven Multi-object Initial Selection

Require: 3D scene S, current viewpoint V, current gaze position $gaze$, current right-hand gesture $gestR$, left-hand gesture $gestL$, fixed number of candidate objects $\#N_c$, importance coefficient α
Ensure: selected target object set OBJ
1: $IMP \leftarrow zeroImportance(S)$
2: **while** $gestL \neq thumbUp$ **do**
3: $OBJ_c \leftarrow \emptyset$
4: $IMP' \leftarrow zeroImportance(S)$
5: **while** $gestR \neq lifting$ and $gestL \neq thumbUp$ **do**
6: $\Delta IMP \leftarrow calInstIMP(S, V, gaze)$
7: $IMP' \leftarrow sumIMP(\Delta IMP, IMP')$
8: $yield(\Delta t)$
9: **end while**
10: $IMP \leftarrow calIMP(IMP, IMP', \alpha)$
11: $OBJ_c \leftarrow sortObjIMP(S, IMP, \#N_c)$
12: $pan \leftarrow visCandPan(pan, OBJ_c)$
13: $clearCandIMP(OBJ_c, IMP)$
14: **while** $gestR = lifting$ and $gestL \neq thumbUp$ **do**
15: $OBJ \leftarrow OBJ \cup gestSelect(gestR, pan)$
16: **end while**
17: **end while**
18: **return** OBJ

1) Spatiotemporally Correlated Object Importance Calculation: A core component of MOA$_s$ is the dynamic calculation of IMP, estimating the likelihood that each object is a target based on the user's visual attention during scene exploration. This calculation is informed by the following observations of user behavior during visual search in highly occluded environments:

Rule 1: users tend to fixate more frequently on target objects than on non-target objects;

Rule 2: in visually dense regions, each object receives less individual attention, potentially prolonging search time;

Rule 3: gaze fixation duration tends to be longer on or near target objects compared to non-target objects.

The above rules summarize the general patterns of visual behavior exhibited by users during multi-object arrangement tasks. Although individual differences in visual behavior exist, our study aims to model object importance by extracting common visual behavior patterns, thereby ensuring the model's applicability across different users. Therefore, we calculate object importance based on these universal rules, ensuring the model is not only generalizable but also maintains good accuracy across various individuals. We define three levels of importance:

Instantaneous Object Importance (ΔIMP): defines the importance score of an object at a specific moment based on Rules 1 and 2.

Temporal Object Importance (IMP'): accumulates ΔIMP over a period within a selection round following Rule 3.

Spatiotemporally Correlated Object Importance (IMP): since IMP' relies only on current-round gaze motion and scene spatial features to assess object importance, it 1) neglects accumulated attention and selection feedback on the same object from previous rounds, and 2) tends to cause a 'short-sighted' effect where the system estimates from scratch each round, resulting in slow convergence and sensitivity to transient noise. IMP performs a weighted summation between the current-round IMP' and historical IMP. This approach maintains responsiveness to current-round gaze signals and scene spatial information while inheriting long-term statistics across selection rounds, thereby stabilizing candidate object pool updates and enabling the system to lock onto true target objects more efficiently.

Returning to Algorithm 1, in IMP calculation step (lines 1-10). We first initialize the IMP values of all objects in S to zero (line 1). Then, we enter the selection round (line 2). For each selection round, we first initialize the candidate object set OBJ_c to an empty set (line 3), and initialize this-round temporal object importance IMP' of all objects in S to zero.

When the user is not in selection mode (line 5), we continuously calculate ΔIMP (line 6) using the function $calInstIMP$. $calInstIMP$ considers the user's effective attentional field, modeled as the foveal region $fovea$ with the eccentricity of 20° [51]. The calculation of $calInstIMP$ is shown in Equation 1:

$$\Delta IMP[obj, t] = e^{-\frac{d}{d_{max}} \cdot \frac{num}{N}} \tag{1}$$

where d is the screen-space Euclidean distance between the object's center and the gaze point $gaze$, d_{max} denotes the maximum radius of the fovea's cross-section at the depth where the object obj projects onto the central gaze axis, num is the number of objects currently within the cone, and N is the total number of objects in the scene S. This equation ensures that importance decreases exponentially with the object's angular distance d from the gaze position, thus modeling the attentional gradient according to Rule 1. Moreover, the rate of decrease is modulated by the density factor (num/N), causing importance to decline sharply when many objects compete for attention within the cone, reflecting attentional dilution as described in Rule 2. Then, the temporal object importance IMP' for the current round is calculated by accumulating these instantaneous values through $sumIMP$ over a period of $\Delta t = 0.1s$ using the $yield$ function (lines 7-8). The Δt accumulation period is designed to balance smooth, real-time operation under typical device performance constraints while capturing meaningful changes in user attention. The calculation in $sumIMP$ is shown in Equation 2:

$$IMP'[obj] = \sum_{t \in T_{accu}} \Delta IMP[obj, t] \cdot \Delta t \tag{2}$$

where T_{accu} represents the set of discrete $0.1s$ time steps.

When the selection mode is activated, IMP is updated by the function $calIMP$ (line 10), as shown in Equation 3:

$$IMP = \alpha \cdot IMP' + (1 - \alpha) \cdot IMP \tag{3}$$

In this update, IMP' from the current round is assigned a weight of α, while the accumulated importance from previous rounds IMP is assigned a weight of $(1 - \alpha)$. This weighted combination ensures the system remains responsive to current user interactions while leveraging long-term historical data. Such an approach facilitates efficient convergence toward target objects, even when they are not fully identified in a single round.

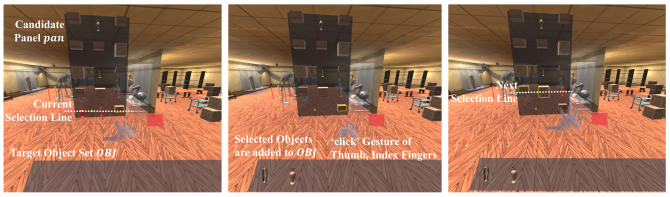
2) Natural Multi-object Initial Selection Mode: This section details the natural multi-object initial selection mode based on IMP, designed for efficiently and conveniently selecting numerous target objects from densely clustered objects in high-occlusion scenes.

Returning to Algorithm 1, we use a right-hand 'lifting' gesture to activate the initial selection mode (lines 11-16). We first identify a candidate object set OBJ_c by sorting objects in descending order of their IMP scores and selecting the top $\#N_c$ items (line 11). This design is motivated by two key considerations: leveraging accumulated user attention data to assign higher priority to the most probable target objects, thereby reducing the user's decision-making burden; limiting the candidate set to $\#N_c$ items to prevent cognitive overload caused by presenting excessive options in complex scenes.

Subsequently, the algorithm uses the function $visCandPan$ to visualize these $\#N_c$ candidate objects in the virtual candidate panel pan (line 12). The function $visCandPan$ serves three key purposes: (a) transforming the unstructured 3D scene search into a structured 2D selection, significantly reducing positioning costs; (b) aggregating distant or occluded targets in a dedicated panel to alleviate visual load; (c) ensuring fair comparison and interaction consistency by normalizing object dimensions based on their longest edges and arranging them uniformly in a grid layout. In pan, the number of candidate objects in each column is w, and in each row is h. We fix the number of visualized candidate objects as 15, that is, $w \cdot h = 15$. w is set to 3, corresponding to thumb, index, and middle finger touch gestures to establish intuitive control. Thus, h is set to 5. Candidate objects are visualized in pan in descending order of their IMP scores, starting from the ergonomically optimal bottom-left corner to minimize hand movement. This layout integrates VR hand motion ergonomic boundaries and the decreasing dexterity gradient from thumb to middle finger to accelerate selection [52].

Next, the function $clearCandIMP$ is used to clear the IMP values of objects remaining on the pan (line 13). This enables a truly iterative selection mechanism: (a) resetting the IMP signals that these objects have already had a full opportunity to be chosen; (b) if the user does not select them, the system infers they do not align with the current intent; (c) clearing them prevents these objects from repeatedly reappearing in subsequent rounds solely based on historical attention, allowing the candidate object set to update in real time during the selection mode. By continually refreshing the candidate object set, the algorithm quickly converges on the actual target objects.

Then, the user enters the target object selection while the user maintains the right-hand 'lifting' gesture, using $gestSelect$ to map intuitive hand movements to selection



(a) (b) (c)

Fig. 2. Process visualization of natural multi-object initial selection mode.

actions (lines 14-16). The details of $gestSelect$ are shown in Fig. 2:

(a) Row Selection: the palm's pitch angle is linearly mapped to the row index on pan. The system divides the comfortable vertical motion range into h segments, allowing users to finely adjust the wrist to precisely lock onto the desired row.

(b) Column Selection: the three-column layout corresponds to clicks with the thumb, index finger, and middle finger. Users tap with a single finger to select an individual object or tap with multiple fingers simultaneously to select multiple objects in the same row at once. Tapped elements are immediately written into the target object set OBJ.

(c) Personalized Comfort Calibration: to accommodate different users' movement range and sensitivity, both $lifting$ and $gestSelect$ need to be customized. This process involves a one-time calibration procedure tailored to individual users. First, the left-hand palm lifting/pitch angle calibration determines the user's comfortable vertical motion range for adapting the $lifting$ gesture. Specifically, the system guides the user to naturally swing their arm up and down five times, capturing the upper and lower angle boundaries of the palm during these movements, and uses the average of all upper and lower angle boundaries to recognize the $lifting$ gesture. Next, the fingertip click calibration sets a dedicated $gestSelect$ trigger threshold for each finger. The user performs five clicks with each of three fingers used for selection. During each click, the system collects five instantaneous speed readings and takes their median as that click's speed value. It then calculates the average of these five median speed values to determine whether to trigger the $gestSelect$ function.

Finally, the entire initial selection step ends when the user clearly indicates completion with a left-hand thumb-up gesture (line 17); the algorithm then returns the identified target object set OBJ for the later manipulation step (line 18).

B. Auxiliary-Structure-guided Multi-object Later Manipulation

After the multi-object initial selection, we perform multi-object later manipulation to manipulate each target object in the target object set to its corresponding target position. We propose the MOA$_m$, which leverages a novel manipulation mode guided by the proposed auxiliary structure to efficiently manipulate numerous target objects.

Given the target object set OBJ, gaze position set for the last 5 frames $GAZE$, current gesture $gest$, structure quantity coefficient β, and the element thickness coefficient γ, Algorithm 2 demonstrates the process of MOA$_m$. MOA$_m$ has two steps: auxiliary structure initialization (line 1) and update (lines 5-6), and multi-modal multi-object later manipulation mode (lines 3-4). Specifically, MOA$_m$ first initializes the auxiliary structure Ω, the temporary obtained target positions array POS', and the temporary variables manipulated object array OBJ' for updating Ω (line 1). Then, it uses the proposed manipulation mode to manipulate each target object obj in OBJ to get the target position pos_t (lines 2-4), add pos_t to POS' and obj to OBJ' (line 5). After that, it optimizes POS' and OBJ', and updates Ω to more efficiently guide later object manipulations until the multi-object later manipulation is completed (lines 6-7). We elaborate on the initialization and update of the auxiliary structure in Section III-B1, and the multi-modal multi-object later manipulation mode in Section III-B2.

Algorithm 2 Auxiliary-Structure-guided Multi-object Later Manipulation

Require: target object set OBJ, gaze position set for the last 5 frames $GAZE$, current gesture $gest$, structure quantity coefficient β, element thickness coefficient γ

1: $\Omega, POS', OBJ' \leftarrow \emptyset, \emptyset, \emptyset$
2: **for** $obj \in OBJ$ **do**
3: $\quad pos_c \leftarrow coarseManipulate(\overline{GAZE}, \Omega, obj)$
4: $\quad pos_t \leftarrow fineManipulate(gest, obj, pos_c)$
5: $\quad POS', OBJ' \leftarrow POS' \cup pos_t, OBJ' \cup obj$
6: $\quad \Omega, POS', OBJ' \leftarrow updateStruct(\Omega, POS', OBJ', \beta, \gamma)$
7: **end for**

1) Auxiliary Structure Construction: Due to the natural regularity of scene design and the goal of reducing user cognitive load and optimizing user experience, in the VR scene with the task of manipulating many target objects to designated target positions, these designated target positions often have spatial associations within the scene [53], [54], such as symmetry, alignment, hierarchical structuring, proximity, accessibility, etc. [35], [55]–[58]. In geometric approximation, line segments can succinctly describe spatial alignment, proximity, and accessibility, and circles can describe spatial symmetry and hierarchical structuring. Furthermore, combining line segments and circles can efficiently achieve the fitting of closed polygons [59]. To efficiently guide multi-object later manipulation while avoiding overfitting issues that arise from flexible position fitting methods like Bezier curves or splines when describing spatial relationships of target positions, the constructed auxiliary structure utilizes two guiding elements-straight lines and circles-to flexibly direct users in manipulating target objects to designated target positions.

The auxiliary structure construction includes two steps: the structure initialization and update. During the initialization of the auxiliary structure Ω, we define the attributes of shapes in the line shape set $LINE$ and the circle shape set CIR included in Ω. $LINE$ consists of multiple line elements for auxiliary multi-object later manipulation, each line l has seven attributes: the centroid $l.o$, the direction vector $l.dir$, the internal target position array $l.POS$, the internal manipulated objects array $l.OBJ$, the accumulated bias $l.bias$, the time index $l.t$, and the thickness $l.th$. The position of l in the 3D scene is determined by $l.o$ and $l.dir$. $l.POS$ stores target positions described by l and $l.bias$, and $l.OBJ$ stores the

internal manipulated objects described by l. $l.bias$ records the sum of the shortest distances between all target positions in $l.POS$ and l. $l.t$ records the last used time of l. $l.th$ defines the thickness of l based on all the target objects on l, ensuring that l has a thickness that provides clear and intuitive guidance. Similarly, CIR is composed of multiple circle elements, with each circle cir defined by its center $cir.o$, the normal vector $cir.norm$, the radius $cir.r$, the internal target position set $cir.POS$, the internal manipulated objects $cir.OBJ$, the accumulated bias $cir.bias$, the time index $cir.t$, and the thickness $cir.th$. The position of cir in the 3D scene is determined by $cir.o$, $cir.norm$, and $cir.r$. $cir.POS$ stores the target positions that cir describes. $cir.OBJ$ stores the internal manipulated objects within cir. $cir.bias$ records the sum of the shortest distances between all target positions in $cir.POS$ and cir. $cir.t$ records the last used time of cir. $cir.th$ defines the thickness of cir based on all the target objects on cir.

Algorithm 3 Auxiliary Structure Update

Require: auxiliary structure Ω, obtained target position array POS', manipulated object array OBJ', structure quantity coefficient β, element thickness coefficient γ
Ensure: updated Ω, POS', OBJ'

1: $\tau_{dis} \leftarrow avg(OBJ'[-1].bbox)$
2: $l, cir \leftarrow inRange(\Omega, POS'[-1], \tau_{dis})$
3: **if** l is $\varnothing$ **then**
4: $\quad \Omega.LINE, POS', OBJ' \leftarrow createLine(\Omega.LINE, POS', OBJ', \gamma)$
5: **else**
6: $\quad l \leftarrow lineFit(l, POS'[-1], OBJ'[-1], \tau_{dis}, \gamma)$
7: **end if**
8: **if** cir is $\varnothing$ **then**
9: $\quad POS', OBJ', \Omega.CIR \leftarrow createCircle(\Omega.CIR, POS', OBJ', \tau_{dis}, \gamma)$
10: **else**
11: $\quad cir \leftarrow cirFit(cir, POS'[-1], OBJ'[-1], \gamma)$
12: **end if**
13: $\Omega \leftarrow eliminate(\Omega, \beta)$
14: **return** Ω, POS', OBJ'

Algorithm 3 shows the process of Ω update. It first calculates the average size of obj's bounding box $obj.bbox$ and regards it as the threshold τ_{dis} to determine whether the object belongs to a specific shape in Ω (line 1). Then, it applies $inRange$ to confirm whether there exists a line l and a circle cir such that the shortest distance between the last obtained target position $POS'[-1]$ and them is less than τ_{dis}, and return l, cir (line 2). If no lines or circles in Ω meet the condition, it uses functions $createLine$ or $createCircle$ to create a new line element l or a new circle element cir into Ω (lines 3-4, 8-9). Otherwise, it updates the attributes of l or cir by $lineFit$ or $cirFit$ (lines 5-7, 10-12). To keep the number of guiding elements in Ω at an appropriate level, it limits the number of guiding elements to β by $eliminate$ (line 13). Specifically, for the elements in Ω, it sorts them by usage time in ascending order and deletes the earliest used elements until the number of elements in Ω is less than or equal to β. Finally, we return the updated auxiliary structure Ω (line 14).

Next, we provide a detailed explanation of the functions $createLine$, $lineFit$, $createCircle$, and $cirFit$.

The pseudocode of $createLine$ is shown in Algorithm 4, it adds a new line element l into $\Omega.LINE$ and optimizes the obtained target position array POS' and the manipulated object array OBJ'. In Algorithm 4, we first initialize the temporary line set L as an empty set (line 1). Since OBJ' stores the manipulated target objects, and POS' records their corresponding positions, we iterate through the indices idx of OBJ' to synchronously access each pair of elements in both (line 2). In each iteration, we first get the current time t to update the time index for the updated line element (line 3). Then, we get the current iterated target position p_c and object o_c (line 4), and the last (newly-added) target position p_e and object o_e in POS' and OBJ' (line 5). We use the function $closetL$ to traverse L and find the line $L[i]$ whose direction $L[i].dir$ is closest to $\overrightarrow{p_e - p_c}$, and set ang_{min} to $|L[i]|.dir - \overrightarrow{p_e - p_c}|$ (line 6). If ang_{min} is less than 5° (line 7), we update the attributes of $L[i]$ using the function $updateLine$ (line 8), as shown in Equation 4:

$$\begin{aligned} L[i].dir &= \frac{\|L[i].POS\| \cdot L[i].dir + \overrightarrow{p_e - p_c}}{\|L[i].POS\| + 1} \\ L[i].bias &= L[i].bias + ang_{min} \\ L[i].OBJ &= L[i].POS \cup o_c \\ L[i].POS &= L[i].POS \cup p_c \\ L[i].t &= t \\ L[i].th &= \frac{\sum_{obj \in L[i].OBJ} avg(obj.bbox)}{\|L[i].POS\| \cdot \gamma} \\ L[i].o &= L[i].POS[0] \end{aligned} \tag{4}$$

where $\|\cdot\|$ is the counting operation. Otherwise, we set $L[i]$ to $\varnothing$ in Equation 4 to generate a new guiding line element l, and add l to L (lines 9-11). After the iteration, we find the optimal line l from L with the maximum number of internal target positions and the minimum bias (line 13), and add it to $\Omega.LINE$ (line 14). After that, we remove all target objects and positions in l from OBJ' and POS' to avoid duplicate fitting (lines 15-16). Finally, we return the updated $\Omega.LINE$, and optimized OBJ' and POS' (line 17).

Algorithm 5 shows the pseudocode of $lineFit$. The basic idea of $lineFit$ is to seek an optimal subset in $l.POS$ to refit the line element l, which not only avoids the fitting deviation caused by invalid target positions within $l.POS$ but also effectively represents the spatial relationship among obtained target positions. To achieve this, in $lineFit$, we first merge the last target object and corresponding target position into $l.POS$ and $l.OBJ$ (lines 1-2). Then, we use the function $optPosSet$ to find the optimal subset of $l.POS$ (line 3). Specifically, we construct a target position pair set $E(pos)$ from the internal target position set $l.POS$, as described in Equation 5:

$$E(pos) = \{\langle p_i, p_j \rangle \, | p_i \in l.POS \; and \; p_j \in l.POS\} \tag{5}$$

Based on the distance threshold τ_{dis}, the neighboring target position set $E(\langle p_i, p_j \rangle)$ can be calculated by Equation 6:

$$E(\langle p_i, p_j \rangle) = \{p_k \in POS | d(p_k, \langle p_i, p_j \rangle) < \tau_{dis}\} \tag{6}$$

Algorithm 4 Create Line

Require: line set $\Omega.LINE$, obtained target position array POS', manipulated object array OBJ', element thickness coefficient γ
Ensure: updated $\Omega.LINE, POS', OBJ'$
1: $L \leftarrow \emptyset$
2: **for** $idx \in range[0, len(POS)]$ **do**
3: $t \leftarrow curTime()$
4: $p_c, o_c \leftarrow POS'[idx], OBJ'[idx]$
5: $p_e, o_e \leftarrow POS'[-1], OBJ'[-1]$
6: $L[i], ang_{min} \leftarrow closestL(L, \overrightarrow{p_e - p_c})$
7: **if** $ang_{min} < 5°$ **then**
8: $L[i] \leftarrow updateLine(L[i], \{o_c, o_e\}, \{p_c, p_e\}, ang_{min}, t, \gamma)$
9: **else**
10: $L \leftarrow L \cup updateLine(\varnothing, \{o_c, o_e\}, \{p_c, p_e\}, 0, t, \gamma)$
11: **end if**
12: **end for**
13: $l \leftarrow optLine(L)$
14: $\Omega.LINE \leftarrow \Omega.LINE \cup l$
15: $POS' \leftarrow POS' - l.POS$
16: $OBJ' \leftarrow OBJ' - l.OBJ$
17: **return** $\Omega.LINE, POS', OBJ'$

Algorithm 5 Fit Line

Require: line element l, last target position pos, last manipulated object obj, distance threshold τ_{dis}, element thickness coefficient γ
Ensure: updated l
1: $l.OBJ \leftarrow l.OBJ \cup obj$
2: $l.POS \leftarrow l.POS \cup pos$
3: $l.POS \leftarrow optPosSet(l.POS, \tau_{dis})$
4: $l.o, l.dir \leftarrow LeastSquare(l.POS)$
5: $l.OBJ \leftarrow corOBJ(l.POS, l.OBJ)$
6: $l.t \leftarrow curTime()$
7: $l.th \leftarrow calThick(l.OBJ, \gamma)$
8: **return** l

where $d(p_i, p_j)$ is used to calculate the Euclidean distance between p_i and p_j. Then, $lineFit$ finds the optimal target position pair $\langle p_i^*, p_j^* \rangle$ with the highest number of neighboring target positions, as shown in Equation 7:

$$\langle p_i^*, p_j^* \rangle = argmax\left(\|E(\langle p_i, p_j \rangle)\|\right) \tag{7}$$

where $optPosSet$ outputs the target position set $\{p_i^*, p_j^*\} \cup E(\langle p_i^*, p_j^* \rangle)$, and sets it to $l.POS$. We use the least square method [60] to calculate the centroid $l.o$ and the direction vector $l.dir$ of l based on $l.POS$ (line 4). Then, $l.OBJ$ is set to all the target objects corresponding to the target positions in $l.POS$ (line 5), $l.t$ is set to the current time (line 6), and $l.th$ is calculated by Equation 8 (line 7):

$$l.th = \frac{\sum_{obj \in l.OBJ} avg(obj.bbox)}{\|l.POS\| \cdot \gamma} \tag{8}$$

where γ is used to optimize the thickness of the guiding element. Finally, we return the updated l (line 8).

The pseudocode of $createCircle$ is shown in Algorithm 6, which updates $\Omega.CIR$, and optimizes POS' and OBJ'. First,

Algorithm 6 Create Circle

Require: circle set $\Omega.CIR$, obtained target position array POS', manipulated object array OBJ', distance threshold τ_{dis}, element thickness coefficient γ
Ensure: updated $\Omega.CIR, POS', OBJ'$
1: $POS_a \leftarrow distAsc(POS', POS'[-1])$
2: $POS_{neb} \leftarrow \{pos_{(k)} \in POS_a | 1 \leq k \leq min(15, \|POS_a\|)\}$
3: $CIR' \leftarrow \emptyset$
4: **for** $\langle i, j, k \rangle \in range(len(POS_{neb}))$ **do**
5: $t \leftarrow curTime()$
6: $cir \leftarrow initCir(\langle i, j, k \rangle, POS_{neb}, OBJ', \tau_{dis}, t, \gamma)$
7: $CIR' \leftarrow CIR' \cup cir$
8: **end for**
9: $c \leftarrow getOptCir(CIR')$
10: $\Omega.CIR \leftarrow \Omega.CIR \cup c$
11: $POS' \leftarrow POS' - c.POS$
12: $OBJ' \leftarrow OBJ' - c.OBJ$
13: **return** $\Omega.CIR, POS', OBJ'$

it calculates the distances between all positions in POS' and the latest position $POS'[-1]$, sorts them in ascending order to obtain POS_a (line 1), and then selects the 15 closest target positions to form the neighboring target position set POS_{neb} (line 2). After that, it creates a temporary circle set CIR' (line 3). It iterates through all triples formed by target positions in POS_{neb} (line 4), creating a circle cir based on each triple (line 6), as shown in Equation 9:

$$\begin{aligned}
&cir.o = \frac{p_i + p_j + p_k}{3} \\
&cir.r = |p_i - cir.o| \\
&cir.POS = \{p | p \in POS_{neb} \ and \ |p - cir.o| < \tau_{dis}\} \\
&cir.OBJ = \{OBJ'[k] | POS'[k] \in POS_{neb}\} \\
&cir.bias = \sum_{p \in cir.POS} |p - cir.o| \\
&cir.t = t \\
&cir.th = \frac{\sum_{obj \in cir.OBJ} avg(obj.bbox)}{\|cir.POS\| \cdot \gamma}
\end{aligned} \tag{9}$$

and adds cir to CIR' (line 7). After the iteration, it selects the optimal circle c from CIR' (line 9), adds it to the circle set CIR' (line 10), and removes the internal target positions in $cir.POS$ and $cir.OBJ$ from POS' and OBJ' to avoid duplicate fitting (lines 11-12). Finally, it returns the updated $\Omega.CIR$, and the optimized POS' and OBJ' (line 13).

Algorithm 7 shows the pseudocode of $cirFit$. The basic idea of the $cirFit$ is to update the circle shape cir to better describe the spatial correlation between the inner target positions in $cir.POS$ and pos_t. Firstly, it merges the last target position pos and the last manipulated object obj into $cir.POS$ and $cir.OBJ$ (lines 1-2). Then, it regards the center $cir.o$ as the average value of $cir.POS$ (line 3), as shown in Equation

Algorithm 7 Fit Circle

Require: circle element cir, last target position pos, last manipulated object obj, element thickness coefficient γ
Ensure: updated circle element cir
1: $cir.POS \leftarrow cir.POS \cup pos$
2: $cir.OBJ \leftarrow cir.OBJ \cup obj$
3: $cir.o \leftarrow avgPOS(cir.POS)$
4: $cir.norm \leftarrow calNorm(cir.POS)$
5: $cir.r \leftarrow LeastSquare(cir.POS)$
6: $cir.t \leftarrow curTime()$
7: $cir.th \leftarrow calThick(cir.OBJ, \gamma)$
8: **return** cir

10:

$$cir.o = (x_o, y_o, z_o)$$
$$s.t. \begin{cases} x_o = \frac{1}{\|cir.POS\|} \sum_{(x_i,y_i,z_i)\in cir.POS} x_i \\ y_o = \frac{1}{\|cir.POS\|} \sum_{(x_i,y_i,z_i)\in cir.POS} y_i \\ z_o = \frac{1}{\|cir.POS\|} \sum_{(x_i,y_i,z_i)\in cir.POS} z_i \end{cases} \quad (10)$$

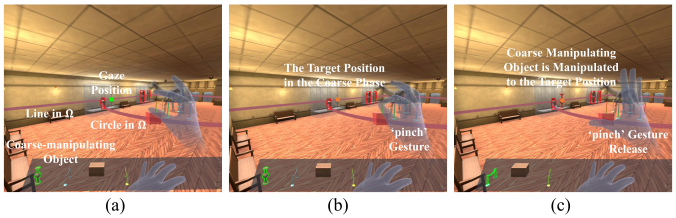
Then, we search for a plane $Ax + By + Cz + D = 0$ in the 3D scene that minimizes the sum of the shortest distances to all target positions in $cir.POS$, i.e., $cir.norm = (A^*, B^*, C^*)$ (line 4). Specifically, it meets Equation 11:

$$(A^*, B^*, C^*) = argmin \sum_{(x_i,y_i,z_i)\in cir.POS} \frac{|Ax_i + By_i + Cz_i + D|}{\sqrt{A^2 + B^2 + C^2}}$$
$$s.t.\ Ax + By + Cz + D = 0 \quad (11)$$

Let $a^* = -\frac{A^*}{C^*}$, $b^* = -\frac{B^*}{C^*}$, $c^* = -\frac{D^*}{C^*}$, Equation 11 can be optimized as Equation 12:

$$\begin{bmatrix} \sum x_i^2 & \sum x_i y_i & \sum x_i \\ \sum x_i y_i & \sum y_i^2 & \sum y_i \\ \sum x_i & \sum y_i & \|cir.POS\| \end{bmatrix} \begin{bmatrix} a^* \\ b^* \\ c^* \end{bmatrix} = \begin{bmatrix} \sum x_i z_i \\ \sum y_i z_i \\ \sum z_i \end{bmatrix} \quad (12)$$

where $(x_i, y_i, z_i) \in cir.POS$. According to the above equation, $cir.norm = (-a^*, -b^*, 1)$, and cir is located in the plane $-a^*x - b^*y + z - d^* = 0$. Based on the distances from all projected points to $cir.o$, $cir.r$ is calculated using the least squares method (line 5). $cir.t$ is set to the current time (line 6), and $cir.th$ is calculated by Equation 8 (line 7). Finally, it returns the updated circle element cir (line 8).

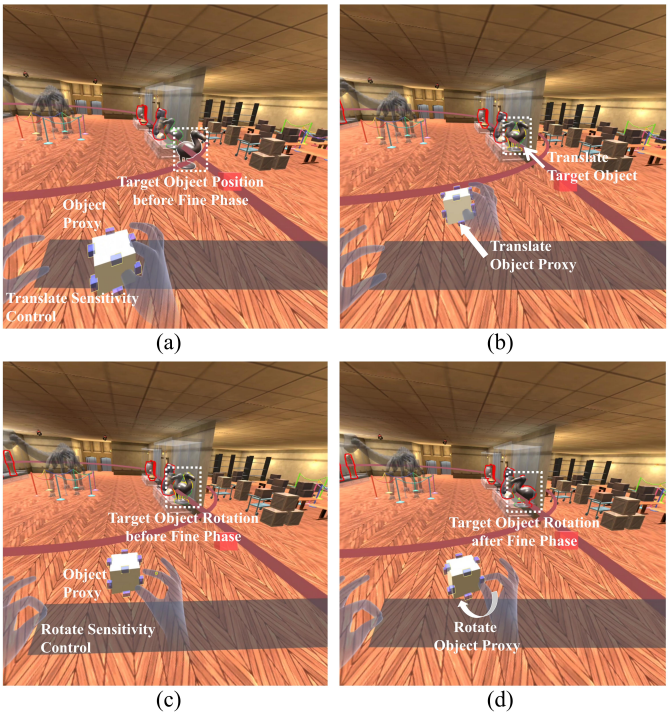


(a) (b) (c)

Fig. 3. Process visualization of the multi-modal multi-object later manipulation mode's coarse phase.

2) Multi-modal Multi-object Later Manipulation Mode: We propose a multi-modal multi-object later manipulation mode based on the user's gaze and gesture according to the guidance of the auxiliary structure. The later manipulation is divided into coarse and fine phases according to Zheng et al. [17]. During the coarse phase, the target object is manipulated to the vicinity of the target position. Then, in the fine phase, the target object is precisely manipulated from the nearby position to the exact target position.

We perform the coarse phase of later manipulation based on the user's gaze and gestures, as illustrated in Fig. 3. First, we select the target object with the highest IMP as the coarse-manipulating object (highlighted by a green outline in OBJ) and define the intersection of the view ray and the scene as the 3D gaze position (marked by a green dot), as shown in Fig. 3 (a). To mitigate eye tracker errors, we calculate the 3D gaze position using the average view ray over the previous 5 frames. Next, the user performs a 'pinch' gesture with the right hand to confirm the 3D gaze position as the coarse-manipulating target position (marked by an orange dot), shown in Fig. 3 (b). After the user releases the 'pinch' gesture, we manipulate the coarse-manipulating object to this target position, as shown in Fig. 3 (c).



(a) (b) (c) (d)

Fig. 4. Process visualization of the multi-modal multi-object later manipulation mode's fine phase.

In the fine phase, we construct an object proxy to enable efficient and convenient translation and rotation of target objects, allowing precise control from the position after the coarse phase to the designated target position for each target object in OBJ, as shown in Fig. 4. The user pinches the object proxy with the right hand to translate and rotate it, while the distance between the left-hand thumb and index finger controls manipulation sensitivity. As shown in Fig. 4 (a), the object proxy appears near the user's right hand. The user pinches the white area of the proxy to perform translation manipulation, with the distance between the left-hand thumb and index finger controlling the translation speed; the greater the distance, the faster the speed. Fig. 4 (b) shows that the user successfully performs translation manipulation during the fine phase guided by Ω. The user pinches the blue axis of the object proxy to

(a) *diningroom* (b) *museum* (c) *garden* (d) *street*

Fig. 5. Visualization of all test scenes in the main user study.

perform rotation manipulation, with the left-hand thumb-index finger distance controlling the rotation speed; the greater the distance, the faster the rotation, as shown in Fig. 4 (c). Fig. 4 (d) visualizes the successful rotation of the target object to the specified target orientation using the object proxy.

IV. Main User Study

The experimental results from user studies 1 and 2 (detailed in Sections 2 and 3 of the supplementary material) show that in the general multi-object arrangement scene, MOA significantly outperforms state-of-the-art controller-free and controller-based methods in task performance, task load, and convenience. In this main study, we further quantify the effects of MOA under different levels of proficiency in VR scenes involving multi-object arrangement tasks of varying complexity. We formulate a hypothesis for the main user study:
H1. Compared to state-of-the-art controller-free and controller-based multi-object arrangement methods, with comparable learning costs, MOA achieves significant improvements in task performance, task load, and convenience across multiple multi-object arrangement scenes of varying complexity.

A. Main User Study Design

Participants. We recruit a new cohort of 24 participants—eleven male and thirteen female—aged 20 to 40. None have participated in the pilot user study or user studies 1 and 2 detailed in the supplementary material. All participants have normal or corrected-to-normal vision, and 12 of them have prior experience using HMD VR applications.

Apparatus. Our system uses a PICO 4 Pro HMD powered by a workstation with a 3.8GHz Intel(R) Core(TM) i7-10700KF CPU, 32GB of RAM, an NVIDIA GeForce GTX 3080Ti graphics card, and an HTC Vive tracker. The resolution of the HMD is 2160×2160 pixels for each eye, and the field-of-view is $40°$. We use the built-in programs of PICO 4 Pro to implement gaze and hand gesture tracking. Our program is developed with C# and HLSL, and is run in Unity 2021.3.8f1.

Test Scene Design. We construct four multi-object arrangement scenes, as shown in Fig. 5, achieving full coverage of multi-task complexity that ranges from 37 to 512 candidate objects, and from 16 to 120 target objects [31], [61]. The $diningroom$ scene includes 37 candidate objects, with 16 target objects. Participants are required to set a table by selecting designed plates, forks, and chopsticks from dense tableware on a countertop and placing them at designated target positions on the dining table, as shown in Fig. 5 (a). The $museum$ scene includes 46 candidate objects, with 17 target objects. Participants arrange exhibits by selecting specific items from many candidate exhibits in a warehouse and placing them at designated target positions within the exhibition hall, as shown in Fig. 5 (b). The $garden$ scene includes 256 candidate objects, with 90 target objects. Participants beautify the garden by selecting target potted plants and floors from two collections with highly occluded objects on both sides, then placing them in specified positions at the garden's center, as shown in Fig. 5 (c). The $street$ scene includes 512 candidate objects, with 128 target objects. Participants arrange a community street by selecting colored trees from the tree collection and colored cars from the car collection, placing trees at designated target positions along both sides of the street and moving cars to specified locations on the street, as shown in Fig. 5 (d).

Condition. According to the experimental results of user studies 1 and 2, in terms of state-of-the-art controller-free methods, Bubble+Object Proxy achieves the best effects in the general multi-object arrangement task. Thus, in the main user study, We compare the proposed MOA with the state-of-the-art controller-free method Bubble+Object Proxy, and the state-of-the-art controller-based method VVIR. Therefore, the main user study includes three method conditions: MOA, $Bubble + Object\ Proxy$, and $VVIR$. The coefficients for MOA are optimized via a pilot study, which is detailed in Section 1 of the supplementary material, establishing the configuration used in the main user study as $MOA = MOA_s(\alpha = 2/3) + MOA_m(\beta = 8, \gamma = 6)$.

Task and Procedure. To assess the learning cost of different conditions, we divide the whole experiment into six sessions, one per day. For each test scene, the task is to select all target objects from the candidate objects in the scene and then manipulate them to designated target positions. To ensure the fairness and rigor of the main user study, we employ a mixed-design experiment, randomly assigning 24 participants to four groups of six, each with three novices and three experienced participants. Each group is uniquely assigned to one of four test scenes, which serve as a between-subjects variable, meaning each participant experiences only one fixed test scene throughout the study. Within each group of six, we implement a fully counterbalanced presentation order of the three method conditions. By generating all possible permutations of these three conditions (resulting in six unique orders), we randomly assign each of the six participants in the group to one of these orders on a one-to-one basis. Each participant consistently follows his/her uniquely assigned order to complete a multi-object arrangement task across six experiment sessions. This multi-layered design systematically mitigates learning and fatigue effects associated with method order, while ensuring observed changes in participant performance over time under consistent experimental conditions. Before the formal experiment begins, we require participants to arrange

three target objects using three different method conditions in *general scene*, with each condition taking approximately 1 minute. Each participant needs to complete three trials per session. After each trial is completed, the participant is required to fill out the NASA-TLX and SUS questionnaires. For each participant, the initial positions of all conditions are fixed (details are demonstrated in the penultimate paragraph of Section 1.1 in the supplementary material), and all candidate objects and target positions are within the visual field. Each participant takes an average of 40 minutes per session. A total of 24 (participants) $\times$ 6 (sessions) $\times$ 3 (method conditions) = 432 trials are collected.

Metrics. We use the objective metrics *total time cost* to quantify the task performance of the multi-object arrangement. The *total time cost* records the time participants spend selecting all target objects and manipulating them to corresponding target positions. we use the standard NASA-TLX questionnaire [62] to evaluate task load. We use the System Usability Scale (SUS) [63] for each method condition to evaluate convenience.

B. Results and Discussion

Before analysis, we use Shapiro-Wilk tests and Q-Q plots to examine the normal distribution of the data, and utilize the ART to transform non-normally distributed data. Then, we conduct ANOVA analyses for all comparisons, and perform Bonferroni post-hoc analyses to examine individual differences between MOA and other conditions.

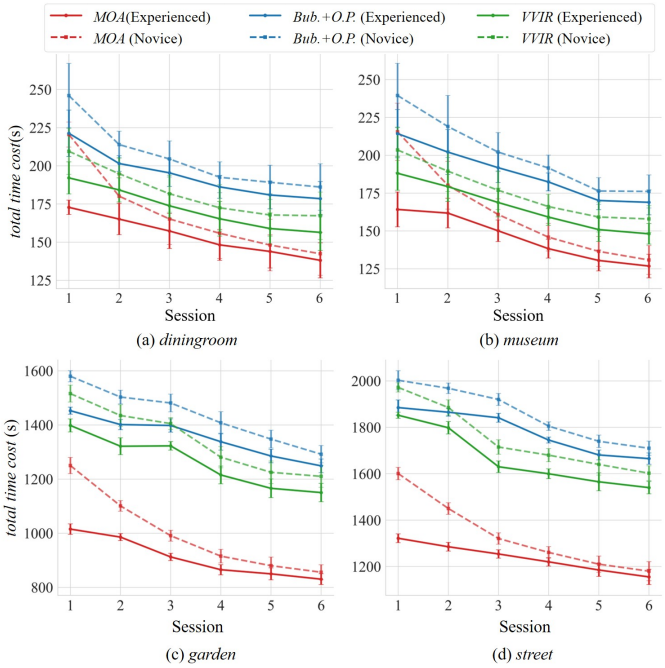


Fig. 6. Plots of average *total time cost* as the function of session under MOA, $Bubble + Object\ Proxy$ ($Bub. + O.P.$), and $VVIR$ in (a) $diningroom$, (b) $museum$, (c) $garden$, and (d) $street$. Error bars represent standard error, and the same applies to subsequent figures.

Task Performance. Fig. 6 shows the average *total time cost* to complete the multi-object arrangement task as a function of sessions for novices and experienced participants under different conditions across four scenes. According to Fig. 6, MOA achieves the lowest average *total time cost* under nearly all sessions. The only exception occurs in session 1, where novices performing the multi-object arrangement task in $diningroom$ and $museum$, which involve few candidate objects, record slightly higher average *total time cost* than $Bubble + Object\ Proxy$, but still outperform $Bubble + Object\ Proxy$. According to participant feedback, this is because the initial learning cost of MOA outweighs its efficiency advantages in $diningroom$ and $museum$ with short task durations. However, in slightly complex scenes involving more than a hundred candidate objects, the significant advantage of MOA in multi-object arrangement is sufficient to completely offset the learning cost, enabling novices to outperform the other two method conditions in session 1. For all sessions, the average *total time cost* of MOA is 639.1, while the average *total time cost* of $Bubble + Object\ Proxy$ and $VVIR$ is 902.0 and 839.5, respectively. The effect test of method conditions under *total time cost* yields the average ($F_{2,40} = 128.80$, $p = 4.44 \times 10^{-10}$, $\eta_p^2 = 0.85$), indicating significant differences among the three conditions under *total time cost*.

TABLE I
POST-HOC ANALYSIS BETWEEN MOA AND OTHER CONDITIONS FOR *total time cost*.

metric	comparison		mean dif.	std. dif.	p-value
total time cost	MOA	$Bubble + Object\ Proxy$	-283.1	95.5	8.0×10^{-3}
		$VVIR$	-225.9	95.5	2.1×10^{-2}

Table I presents the post-hoc statistical results comparing MOA with the other two conditions using the Bonferroni method for *total time cost* across all sessions. The average *total time cost* of MOA for all sessions is lower than the corresponding values for $Bubble + Object\ Proxy$ and $VVIR$ in all four scenes. The p-values for comparisons between MOA and each of $Bubble + Object\ Proxy$ and $VVIR$ are both less than 0.05, indicating significant differences in *total time cost*. These results indicate that, compared to state-of-the-art controller-free and controller-based methods at a similar learning cost, MOA demonstrates superior task performance across all scenes, covering both low complexity (46 candidate objects with 17 target objects) and high complexity (512 candidate objects with 128 target objects). Therefore, we conclude **Conclusion 1**: In all sessions, compared to both state-of-the-art controller-free and controller-based multi-object arrangement methods, MOA demonstrates significant improvements in task performance across all multi-target arrangement scenes with varying task complexities.

Task Load. Regarding task load, Fig. 7 shows the NASA-TLX total score progression across sessions under these conditions for both novices and experienced participants. Especially in $garden$ and $street$ containing over 100 candidate objects, MOA demonstrates greater advantages. As shown in Fig. 7, MOA consistently yields the lowest perceived workload for both novice and experienced participants across all scenes and sessions, outperforming $Bubble + Object\ Proxy$ and $VVIR$. The effect test for three conditions in NASA-TLX total score

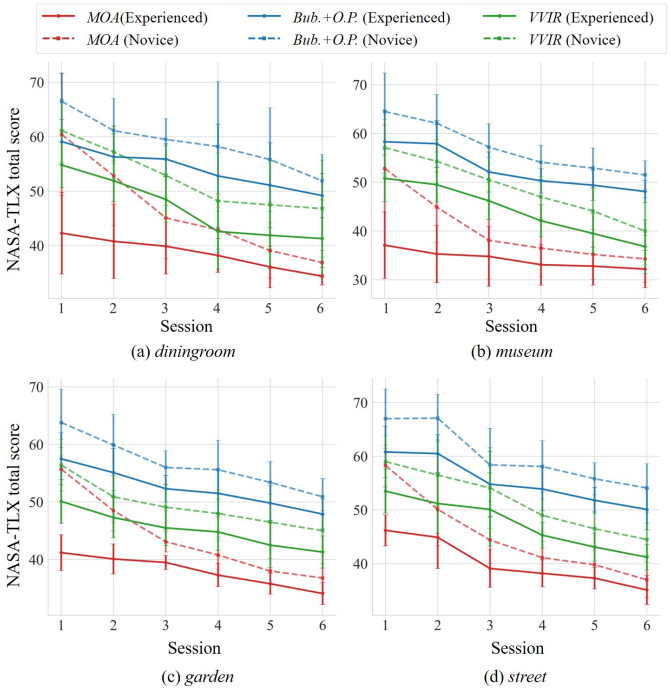


Fig. 7. Plots of the NASA-TLX total score as the function of session under different conditions in (a) $diningroom$, (b) $museum$, (c) $garden$, and (d) $street$.

TABLE II
MEAN ± SD SCORES OF EACH QUESTION IN NASA-TLX QUESTIONNAIRE FOR ALL SESSIONS UNDER DIFFERENT CONDITIONS IN THE MAIN USER STUDY.

	Mean ± SD NASA-TLX scores		
QID	MOA	$Bubble + Object\ Proxy$	$VVIR$
Q1	24.1 ± 6.8	32.5 ± 11.1	40.9 ± 13.5
Q2	50.7 ± 10.9	56.9 ± 9.5	61.8 ± 10.2
Q3	55.9 ± 10.6	63.1 ± 9.1	64.2 ± 8.8
Q4	27.8 ± 6.5	39.6 ± 13.2	54.7 ± 15.4
Q5	42.4 ± 10.1	51.3 ± 12.0	56.1 ± 10.8
Q6	29.3 ± 9.8	36.8 ± 10.1	48.0 ± 16.2
TOTAL	38.4 ± 5.5	46.7 ± 6.2	54.3 ± 5.9

yields ($F_{2,40} = 35.46$, $p = 2.12\times10^{-8}$, $\eta_p^2 = 0.61$), indicating significant differences among the three conditions across all sessions in task load.

Table II presents the specific scores for each question in the NASA-TLX questionnaire under different conditions, and the corresponding post-hoc analysis results are shown in Table III. Across all questionnaire items, MOA's scores show a significant advantage compared to the other two conditions. Compared to $Bubble + Object\ Proxy$, MOA shows a score decrease of 10.9-29.8% in Q1-Q6. Compared to $VVIR$, the decrease ranges from 12.9-49.2%. Notably, for mental demands (Q1), efforts (Q4), and frustrations (Q6), MOA's scores decrease by more than 20% relative to the other two conditions. According to participant feedback, during the multi-object arrangement task in all scenes, in the initial selection, MOA_s reduces mental demands (Q1) and required efforts (Q4) by intelligently predicting and prioritizing target objects. The controller-free selection mode further simplifies the operation. Subsequently, in the later manipulation step, the auxiliary structure of the MOA_m simplifies the multi-object coarse manipulation. This process virtually eliminates the frustrations (Q6) arising from repeated misalignments and again reduces the level of efforts (Q4). These core experiential improvements consequently lead to advantages in other dimensions: a more fluid and efficient workflow alleviates temporal demands (Q3), fewer physical actions lower the physical demands (Q2), and a higher success rate leads to greater user satisfaction with their own task performance (Q5). As a result, the overall task load is comprehensively reduced. Thus, MOA achieves significantly lower NASA-TLX scores than $Bubble + Object\ Proxy$ and $VVIR$. Therefore, we obtain **Conclusion 2**: In all sessions, compared with state-of-the-art controller-free and controller-based multi-object arrangement methods, MOA significantly reduces task load across all multi-target arrangement scenes with varying task complexities.

TABLE III
POST-HOC ANALYSIS BETWEEN MOA AND OTHER CONDITIONS FOR NASA-TLX TOTAL SCORE.

metric	comparison		mean dif.	std. dif.	p-value
Q1	MOA	$Bubble + Object\ Proxy$	-8.4	2.2	1.4×10^{-4}
		$VVIR$	-16.8		8.2×10^{-13}
Q2	MOA	$Bubble + Object\ Proxy$	-6.2	1.9	1.0×10^{-3}
		$VVIR$	-11.1		1.6×10^{-8}
Q3	MOA	$Bubble + Object\ Proxy$	-7.2	1.8	6.8×10^{-5}
		$VVIR$	-8.3		3.5×10^{-6}
Q4	MOA	$Bubble + Object\ Proxy$	-11.8	2.5	2.3×10^{-6}
		$VVIR$	-26.9		1.2×10^{-21}
Q5	MOA	$Bubble + Object\ Proxy$	-8.9	2.1	2.1×10^{-5}
		$VVIR$	-13.7		8.9×10^{-10}
Q6	MOA	$Bubble + Object\ Proxy$	-7.5	2.4	2.0×10^{-3}
		$VVIR$	-18.7		4.7×10^{-13}
TOTAL	MOA	$Bubble + Object\ Proxy$	-9.1	0.9	5.3×10^{-20}
		VVIR	-17.0		1.2×10^{-52}

Convenience. In terms of convenience, Fig. 8 shows changes in SUS total scores under different conditions as sessions progress for both novices and experienced participants. As shown in Fig. 8, despite comparable SUS scores between MOA and $Bubble + Object\ Proxy$ for novices in the $diningroom$ scene during session 1, which arise from the learning cost of controller-free interaction over a short task duration, the SUS scores for MOA are better than those of the other two methods for both novice and experienced participants in all other sessions. The effect test for three conditions in SUS total score yields ($F_{2,40} = 76.31$, $p = 4.63 \times 10^{-12}$, $\eta_p^2 = 0.77$), indicating significant differences among the three conditions in convenience.

Table IV-B details the SUS results across these conditions.

The average SUS score of MOA is 84.8, while the average SUS scores of $Bubble + Object\ Proxy$ and $VVIR$ are 77.8 and 69.4, respectively. The SUS total score of MOA is better than that of $Bubble + Object\ Proxy$ and $VVIR$. According to participant feedback, MOA effectively simplifies multi-object arrangement tasks through its effective target object selection mechanism and the auxiliary-guided manipulation mode. Participants generally find MOA easy to learn and perceive the interaction experience as highly coherent and consistent. Consequently, MOA achieves the highest SUS score, outperforming both $Bubble + Object\ Proxy$ and $VVIR$.

Table IV presents post-hoc statistical results comparing MOA with the other two method conditions for SUS total scores. The p-values indicate that MOA achieves significantly higher SUS total scores than those of $Bubble+Object\ Proxy$ and $VVIR$. Therefore, we obtain **Conclusion 3**: Compared with state-of-the-art controller-free and controller-based multi-object arrangement methods, MOA significantly improves convenience across all multi-target arrangement scenes with varying task complexities under similar learning costs. Thus, based on **Conclusions 1**, **2**, and **3**, the results support **H1**.

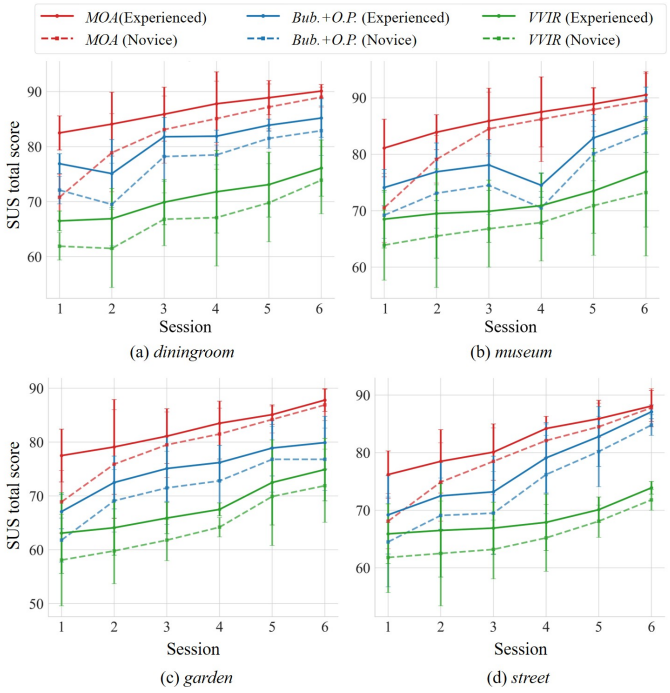


Fig. 8. Plots of the SUS total score as the function of session under different conditions in (a) $diningroom$, (b) $museum$, (c) $garden$, and (d) $street$.

Learning Cost. We discuss the effects of learning cost when performing multi-object arrangement tasks using three method conditions. Among the three method conditions, as shown in Figs. 6-8, MOA exhibits clear overall advantages over $total\ time\ cost$, NASA-TLX total score, and SUS total score during the early sessions (sessions 1-3). The sole exception is a slightly higher $total\ time\ cost$ for novice participants in $diningroom$ and $museum$ during session 1, as the short task durations due to fewer candidate objects do not compensate for participant adaptability in MOA. The post-hoc analysis confirms the statistical significance of MOA's superiority. During the early sessions, for $total\ time\ cost$, post-hoc tests reveal that MOA is significantly better than both $Bubble + Object\ Proxy$ ($p = 1.0 \times 10^{-9}$) and $VVIR$ ($p = 1.0 \times 10^{-9}$). Similar significant advantages are found for NASA-TLX total score (vs. $Bubble + Object\ Proxy$: $p = 1.0 \times 10^{-7}$, vs. $VVIR$: $p = 1.0 \times 10^{-7}$), and SUS total score (vs. $Bubble+Object\ Proxy$: $p = 1.0 \times 10^{-11}$; vs. $VVIR$: $p = 1.0 \times 10^{-11}$) during the early sessions. Thus, we derive **Conclusion 4**: For novices, a single training session involving the multi-object arrangement task with dozens of candidate objects is sufficient to achieve significantly better task performance and user experience when using MOA compared to $Bubble + Object\ Proxy$ and $VVIR$.

Fig. 9 plots the learning curves of $total\ time\ cost$, NASA-TLX score, and SUS score for MOA as the function of the session. As shown in Fig. 9, NASA-TLX total scores continuously decrease and SUS total scores steadily increase during the later sessions (sessions 4-6). This is because participants become more proficient with MOA's interaction mode as the

± SD scores of each question in SUS under different conditions in the main user study.

QID	Mean ± SD SUS scores		
	MOA	$Bubble$ + $Object\ Proxy$	$VVIR$
Q1	4.0 ± 0.8	3.5 ± 0.7	3.6± 0.8
Q2	1.6 ± 0.7	2.0 ± 0.6	2.2 ± 0.8
Q3	4.0 ± 0.8	3.4 ± 0.8	3.4 ± 0.9
Q4	1.6 ± 0.6	2.0 ± 0.8	2.5 ± 0.7
Q5	4.2 ± 0.7	3.9 ± 0.6	3.6 ± 0.8
Q6	1.2 ± 0.5	1.4 ± 0.5	2.0 ± 0.8
Q7	4.4 ± 0.7	4.3 ± 0.7	4.2 ± 0.7
Q8	1.1± 0.4	1.4 ± 0.8	2.0 ± 0.9
Q9	4.3 ± 0.6	4.2 ± 0.7	3.9 ± 0.8
Q10	1.1 ± 0.4	1.3 ± 0.6	1.9 ± 0.7
TOTAL	84.8 ± 6.8	77.8 ± 6.6	69.4 ± 6.3

TABLE IV
POST-HOC ANALYSIS BETWEEN MOA AND OTHER CONDITIONS FOR SUS TOTAL SCORE.

metric	comparison		mean dif.	std. dif.	p (avg.)
SUS	MOA	$Bubble$ +$Object\ Proxy$	7.5	1.0	1.8×10^{-8}
		$VVIR$	16.3	1.0	6.3×10^{-41}

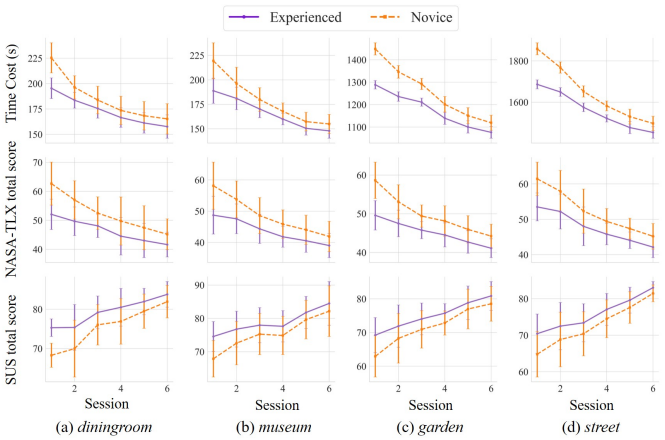


Fig. 9. Learning curves for $total\ time\ cost$, NASA-TLX total score, and SUS total score, comparing Novice and Experienced participants across six sessions in (a) $diningroom$, (b) $museum$, (c) $garden$, and (d) $street$.

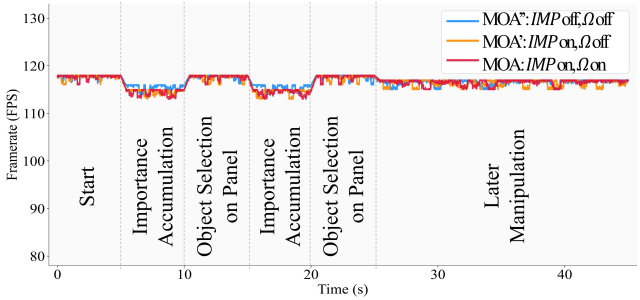
session progresses. However, *total time cost* of MOA tends to stabilize in sessions 4-6. In contrast, as shown in Fig. 6, although the other two method conditions show continuous improvement, their performance levels remain below that of MOA. According to the ANOVA analysis on *total time cost* of MOA, the effect test for sessions 3, 4, and 5 yields ($F_{2,21} = 0.01$, $p = 0.99$, $\eta_p^2 = 1 \times 10^{-3}$), indicating no significant differences. The ANOVA analysis on MOA's *total time cost* in the later sessions shows that a performance plateau of MOA is reached in session 4. Therefore, we obtain **Conclusion 5**: After a brief three-session training period, users can quickly reach the performance plateau when using MOA for multi-object arrangement tasks. Based on **Conclusions 4** and **5**, we regard that compared to state-of-the-art methods, MOA exhibits a steeper learning curve, enabling novices to achieve superior task performance and user experience after a brief training in the multi-object arrangement task.

Proficiency Effects. Figs. 6-8 illustrate the impact of learning effects on task performance and user experience. The experimental data demonstrate that under all test conditions, the task performance and user experience metrics of experienced participants (solid lines) consistently outperform those of novice participants (dashed lines).

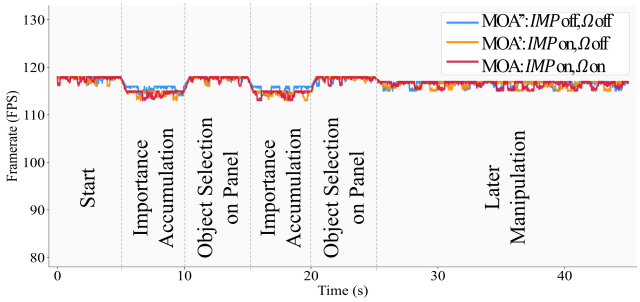
In terms of MOA, as shown in Fig. 9, although experienced participants generally outperform novices initially, the gap in all metrics narrows rapidly with practice. A post-hoc analysis confirms the statistical significance of MOA's superiority. For the post-hoc test of *total time cost*, the p-value between novices and experienced participants across all sessions under MOA is 0.058, indicating that MOA enables novices to quickly match the efficiency of experienced participants. However, significant differences persist in their NASA-TLX and SUS scores. This discrepancy suggests that while novices' performance outcomes (*total time cost*) can rival those of experienced participants, achieving these outcomes demands greater cognitive resources and mental focus. This extra effort to keep up leads to a higher perceived workload.

Running Performance. To further evaluate the real-time performance of the proposed method, Fig. 10 visualizes frame rates under three configurations: MOA with both IMP computation and Ω guidance disabled (MOA": IMP off, Ω off), MOA with only IMP computation enabled (MOA': IMP on, Ω off), and the full MOA (MOA: IMP on, Ω on). The results show that, compared to MOA", the performance of the full MOA decreases by only approximately 0.33% on average across tested scenes. In all scenes, the average frame rates of the full MOA exceed 90 FPS, satisfying the requirements of an immersive interactive experience. Compared to MOA", the average frame rates of MOA' drop by 0.22–0.66 FPS across scenes, indicating that even in complex scenes with up to 512 candidate objects, the IMP calculation takes only 0.07–0.25 ms, causing a negligible impact on the interaction experience.

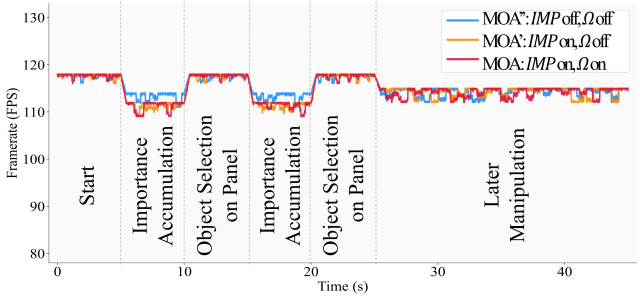
This low computational cost results from optimized computational strategies. Specifically, the IMP calculation has a complexity of $O(N_{cone})$, processing only the N_{cone} objects within the user's view cone every $\Delta t = 0.1\ s$ during user observation, ensuring high speed. The Ω is updated only when an object is manipulated, rather than continuously. Its fitting operations are localized to a small number of structures, with their total count bounded by a constant β, involving only objects directly associated with these specific structures, thereby avoiding extensive computations across the entire scene.



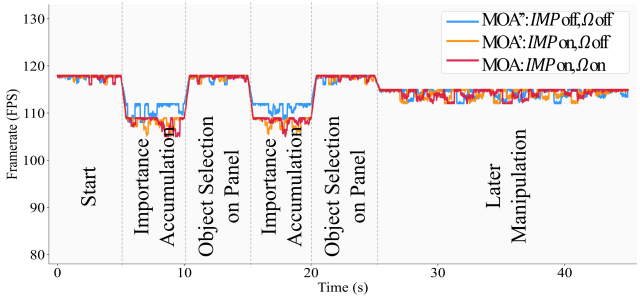
(a) FPS during performing multi-object arrangement in *diningroom*



(b) FPS during performing multi-object arrangement in *museum*



(c) FPS during performing multi-object arrangement in *garden*



(d) FPS during performing multi-object arrangement in *street*

Fig. 10. Plots of the FPS in performing the multi-object arrangement task under different settings of MOA in (a) *diningroom*, (b) *museum*, (c) *garden*, and (d) *street*.

V. Conclusion, Limitation, and Future Work

We have proposed the MOA method that achieves fast and convenient multi-object arrangement in complex VR scenes

with dense candidate objects. Compared to both state-of-the-art controller-free and controller-based multi-object arrangement approaches, MOA achieves significantly-improved task performance, task load, and convenience in multi-object arrangement scenes that contain hundreds of highly occluded objects needed to be arranged.

Although MOA performs well compared with state-of-the-art multi-object arrangement methods, there are some limitations. MOA currently predicts the distribution of target objects based on user attention. It cannot accurately estimate the probability distribution of target objects in regions outside the user's field of view. Furthermore, although MOA demonstrates significant advantages in large-scale object arrangement tasks across diverse indoor and outdoor scenes, the interaction requirements in specific multi-object arrangement scenarios remain inadequately addressed. For instance, in virtual assembly scenarios, users must simultaneously manipulate objects of vastly different sizes from a fixed position, requiring both precise manipulation of tiny screws and large components. Thus, another possible area of future work focuses on exploring a multi-scale, adaptive, multi-object arrangement method based on user intent. This approach dynamically generates a fused view that integrates global context with local focus by predicting user operational intent in real-time. It enables precise manipulation of tiny objects through magnified local views without requiring physical movement, while maintaining awareness of the overall scene, ultimately achieving efficient multi-object arrangement of cross-scale objects.

ACKNOWLEDGMENT

This work is supported by the National Natural Science Foundation of China through Project 62402231, 92473205 and 62302231; and in part by the Natural Science Foundation of the Jiangsu Higher Education Institutions of China 24KJB520027, Jiangsu Province Youth Science and Technology Talent Support Project JSTJ-2025-135, Jiangsu Province Science and Technology Achievement Transformation Special Fund Project BA2022026, and Nanjing University of Posts and Telecommunications Talent Introduction and Research Launch Fund NY224027.

REFERENCES

[1] D. Peeters, "A standardized set of 3-d objects for virtual reality research and applications," *Behavior research methods*, vol. 50, pp. 1047–1054, 2018.

[2] D. Fox, S. S. Y. Park, A. Borcar, A. Brewer, and J. Yang, "Element selection of three-dimensional objects in virtual reality," in *Virtual, Augmented and Mixed Reality: Interaction, Navigation, Visualization, Embodiment, and Simulation: 10th International Conference, VAMR 2018, Held as Part of HCI International 2018, Las Vegas, NV, USA, July 15-20, 2018, Proceedings, Part I 10*. Springer, 2018, pp. 13–29.

[3] S. Jayaram, H. I. Connacher, and K. W. Lyons, "Virtual assembly using virtual reality techniques," *Computer-aided design*, vol. 29, no. 8, pp. 575–584, 1997.

[4] B. Korves and M. Loftus, "Designing an immersive virtual reality interface for layout planning," *Journal of Materials Processing Technology*, vol. 107, no. 1-3, pp. 425–430, 2000.

[5] D. Weidlich, L. Cser, T. Polzin, D. Cristiano, and H. Zickner, "Virtual reality approaches for immersive design," *CIRP annals*, vol. 56, no. 1, pp. 139–142, 2007.

[6] M. W. Kapell and A. B. Elliott, *Playing with the past: Digital games and the simulation of history*. Bloomsbury Publishing USA, 2013.

[7] D. Lee, K. Baek, J. Lee, and H. Lim, "A development of virtual reality game utilizing kinect, oculus rift and smartphone," *International Journal of Applied Engineering Research*, vol. 11, no. 2, pp. 829–833, 2016.

[8] B. E. Riecke, J. J. LaViola Jr, and E. Kruijff, "3d user interfaces for virtual reality and games: 3d selection, manipulation, and spatial navigation," in *ACM SIGGRAPH 2018 Courses*, 2018, pp. 1–94.

[9] H. S. Rejeki, H. Humaedi, and A. Ardiansyah, "Developing manipulative basic movement learning model based on traditional games in elementary schools," *Al-Ta lim Journal*, vol. 29, no. 1, pp. 78–83, 2022.

[10] D. A. Bowman, D. B. Johnson, and L. F. Hodges, "Testbed evaluation of virtual environment interaction techniques," in *Proceedings of the ACM symposium on Virtual reality software and technology*, 1999, pp. 26–33.

[11] K. Pfeuffer, B. Mayer, D. Mardanbegi, and H. Gellersen, "Gaze+ pinch interaction in virtual reality," in *Proceedings of the 5th symposium on spatial user interaction*, 2017, pp. 99–108.

[12] D. Yu, X. Lu, R. Shi, H.-N. Liang, T. Dingler, E. Velloso, and J. Goncalves, "Gaze-supported 3d object manipulation in virtual reality," in *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, 2021, pp. 1–13.

[13] 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*, 2022, pp. 1–6.

[14] S. Park, S. Kim, and J. Park, "Select ahead: efficient object selection technique using the tendency of recent cursor movements," in *Proceedings of the 10th asia pacific conference on Computer human interaction*, 2012, pp. 51–58.

[15] 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.

[16] 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*, 2023, pp. 1–14.

[17] Q. Zheng, L. Wang, W. Ke, and S. K. Im, "Vvir-om: Efficient object manipulation in vr with variable virtual interaction region," *International Journal of Human–Computer Interaction*, pp. 1–14, 2023.

[18] S. Frees and G. D. Kessler, "Precise and rapid interaction through scaled manipulation in immersive virtual environments," in *IEEE Proceedings. VR 2005. Virtual Reality, 2005.* IEEE, 2005, pp. 99–106.

[19] 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*, vol. 26, no. 12, pp. 3402–3413, 2020.

[20] T. T. H. Nguyen, T. Duval, and C. Pontonnier, "A new direct manipulation technique for immersive 3d virtual environments," in *ICAT-EGVE 2014: the 24th International Conference on Artificial Reality and Telexistence and the 19th Eurographics Symposium on Virtual Environments*, 2014, p. 8.

[21] P. C. Gloumeau, W. Stuerzlinger, and J. Han, "Pinnpivot: Object manipulation using pins in immersive virtual environments," *IEEE transactions on visualization and computer graphics*, vol. 27, no. 4, pp. 2488–2494, 2020.

[22] X. Liu, L. Wang, S. Luan, X. Shi, and X. Liu, "Distant object manipulation with adaptive gains in virtual reality," in *2022 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*. IEEE, 2022, pp. 739–747.

[23] 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)*. IEEE, 2021, pp. 216–225.

[24] L. Wang, J. Wu, X. Yang, and V. Popescu, "Vr exploration assistance through automatic occlusion removal," *IEEE transactions on visualization and computer graphics*, vol. 25, no. 5, pp. 2083–2092, 2019.

[25] J. Wu, L. Wang, H. Zhang, and V. Popescu, "Quantifiable fine-grain occlusion removal assistance for efficient vr exploration," *IEEE Transactions on Visualization and Computer Graphics*, vol. 28, no. 9, pp. 3154–3167, 2021.

[26] S. Bhowmick, N. Biswas, P. C. Kalita, and K. Sorathia, "Wow! i have tiny hands: Design and evaluation of adaptive virtual hands for small object selection within arms length in dense virtual environment," in *Extended Abstracts of the 2023 CHI Conference on Human Factors in Computing Systems*, 2023, pp. 1–6.

[27] 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)*. IEEE, 2023, pp. 1221–1230.

[28] H. Jiang, D. Weng, X. Dongye, L. Luo, and Z. Zhang, "Commonsense knowledge-driven joint reasoning approach for object retrieval in virtual reality," *ACM Transactions on Graphics (TOG)*, vol. 42, no. 6, pp. 1–18, 2023.
[29] 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*, vol. 176, p. 103045, 2023.
[30] 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*, 1995, pp. 265–272.
[31] 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)*. IEEE, 2023, pp. 460–470.
[32] 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)*. IEEE, 2011, pp. 67–70.
[33] L. Wang, X. Liu, and X. Li, "Vr collaborative object manipulation based on viewpoint quality," in *2021 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*. IEEE, 2021, pp. 60–68.
[34] R. Li, A. Jabri, T. Darrell, and P. Agrawal, "Towards practical multi-object manipulation using relational reinforcement learning," in *2020 ieee international conference on robotics and automation (icra)*. IEEE, 2020, pp. 4051–4058.
[35] 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*, 2022, pp. 1–11.
[36] X. Li, J.-D. Wang, J. J. Dudley, and P. O. Kristensson, "Swarm manipulation in virtual reality," in *Proceedings of the 2023 ACM Symposium on Spatial User Interaction*, 2023, pp. 1–11.
[37] J. Wu, L. Wang, S. K. Im, and C. T. Lam, "Eeba: Efficient and ergonomic big-arm for distant object manipulation in vr," p. 103273, 2024.
[38] P. Song, W. B. Goh, W. Hutama, C.-W. Fu, and X. Liu, "A handle bar metaphor for virtual object manipulation with mid-air interaction," in *Proceedings of the SIGCHI conference on human factors in computing systems*, 2012, pp. 1297–1306.
[39] R. Arora, R. H. Kazi, D. M. Kaufman, W. Li, and K. Singh, "Magicalhands: Mid-air hand gestures for animating in vr," in *Proceedings of the 32nd annual ACM symposium on user interface software and technology*, 2019, pp. 463–477.
[40] D. Dewez, L. Hoyet, A. Lecuyer, and F. Argelaguet Sanz, "Towards 'avatar-friendly' 3d manipulation techniques: Bridging the gap between sense of embodiment and interaction in virtual reality," in *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, 2021, pp. 1–14.
[41] E. Bozgeyikli and L. L. Bozgeyikli, "Evaluating object manipulation interaction techniques in mixed reality: Tangible user interfaces and gesture," in *2021 IEEE Virtual Reality and 3D User Interfaces (VR)*. IEEE, 2021, pp. 778–787.
[42] S.-H. Lee, T. Jin, J. H. Lee, and S.-H. Bae, "Wiresketch: bimanual interactions for 3d curve networks in vr," in *Adjunct Proceedings of the 35th Annual ACM Symposium on User Interface Software and Technology*, 2022, pp. 1–3.
[43] 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.
[44] D. L. Chen, R. Balakrishnan, and T. Grossman, "Disambiguation techniques for freehand object manipulations in virtual reality," in *2020 IEEE conference on virtual reality and 3D user interfaces (VR)*. IEEE, 2020, pp. 285–292.
[45] M. Moran-Ledesma, O. Schneider, and M. Hancock, "User-defined gestures with physical props in virtual reality," *Proceedings of the ACM on Human-Computer Interaction*, vol. 5, no. ISS, pp. 1–23, 2021.
[46] S. Pei, A. Chen, J. Lee, and Y. Zhang, "Hand interfaces: Using hands to imitate objects in ar/vr for expressive interactions," in *Proceedings of the 2022 CHI conference on human factors in computing systems*, 2022, pp. 1–16.
[47] S. Zhang, Y. Liu, F. Song, D. Yu, Z. Bo, and Z. Zhang, "The effect of audiovisual spatial design on user experience of bare-hand interaction in vr," *International Journal of Human–Computer Interaction*, vol. 40, no. 11, pp. 2796–2807, 2024.
[48] X. Meng, W. Xu, and H.-N. Liang, "An exploration of hands-free text selection for virtual reality head-mounted displays," in *2022 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*. IEEE, 2022, pp. 74–81.
[49] D. Yu, Q. Zhou, T. Dingler, E. Velloso, and J. Goncalves, "Blending on-body and mid-air interaction in virtual reality," in *2022 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*. IEEE, 2022, pp. 637–646.
[50] G. Lee, J. Healey, and D. Manocha, "Vrdoc: Gaze-based interactions for vr reading experience," in *2022 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*. IEEE, 2022, pp. 787–796.
[51] H. Strasburger, I. Rentschler, and M. Jüttner, "Peripheral vision and pattern recognition: A review," *Journal of vision*, vol. 11, no. 5, pp. 13–13, 2011.
[52] D. Kee and W. Karwowski, "The boundaries for joint angles of isocomfort for sitting and standing males based on perceived comfort of static joint postures," *Ergonomics*, vol. 44, no. 6, pp. 614–648, 2001.
[53] D. A. Bowman and R. P. McMahan, "Virtual reality: how much immersion is enough?" *Computer*, vol. 40, no. 7, pp. 36–43, 2007.
[54] R. A. Ruddle, S. J. Payne, and D. M. Jones, "Navigating large-scale virtual environments: what differences occur between helmet-mounted and desk-top displays?" *Presence*, vol. 8, no. 2, pp. 157–168, 1999.
[55] A. Agnès, F. Sylvain, R. Vanukuru, and S. Richir, "Studying the effect of symmetry in team structures on collaborative tasks in virtual reality," *Behaviour & Information Technology*, vol. 42, no. 14, pp. 2467–2475, 2023.
[56] W. Hunt, M. Mara, and A. Nankervis, "Hierarchical visibility for virtual reality," *Proceedings of the ACM on Computer Graphics and Interactive Techniques*, vol. 1, no. 1, pp. 1–18, 2018.
[57] R. Kazlauskaitė, "Knowing is seeing: distance and proximity in affective virtual reality history," *Rethinking History*, vol. 26, no. 1, pp. 51–70, 2022.
[58] M. Teófilo, V. F. Lucena, J. Nascimento, T. Miyagawa, and F. Maciel, "Evaluating accessibility features designed for virtual reality context," in *2018 IEEE international conference on consumer electronics (ICCE)*. IEEE, 2018, pp. 1–6.
[59] G. Maier and G. Pisinger, "Approximation of a closed polygon with a minimum number of circular arcs and line segments," *Computational Geometry*, vol. 46, no. 3, pp. 263–275, 2013.
[60] Å. Björck, "Least squares methods," *Handbook of numerical analysis*, vol. 1, pp. 465–652, 1990.
[61] 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*, 2021, pp. 1–20.
[62] S. G. Hart and L. E. Staveland, "Development of nasa-tlx (task load index): Results of empirical and theoretical research," in *Advances in psychology*. Elsevier, 1988, vol. 52, pp. 139–183.
[63] J. Brooke *et al.*, "Sus-a quick and dirty usability scale," *Usability evaluation in industry*, vol. 189, no. 194, pp. 4–7, 1996.

Xuehuai Shi received his Ph.D. degree from the State Key Laboratory of Virtual Reality Technology and Systems, School of Computer Science and Engineering, Beihang University, Beijing, China. He is currently a lecturer at the School of Computer Science, Nanjing University of Posts and Telecommunications. His research interests include virtual reality, human-computer interaction, augmented reality, and real-time rendering.

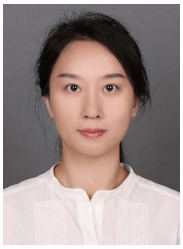
Yuhan Duan is a junior undergraduate student majoring in Beihang University, Beijing, China. She passionates about computer science and has a keen interest in artificial intelligence, machine learning, and data science. She has actively participated in various research projects and competitions, committed to solving real-world problems through technological innovation.

Ziteng Wang is currently pursuing a major in the State Key Laboratory of Virtual Reality Technology and Systems, School of Computer Science and Engineering, Beihang University, Beijing, China. His research interests include virtual reality, human-computer interaction, and augmented reality.

Jian Wu received his Ph.D. degree from the State Key Laboratory of Virtual Reality Technology and Systems, School of Computer Science and Engineering, Beihang University, Beijing, China. He is currently an assistant professor at the School of Computer Science and Engineering, Beihang University, Beijing, China. His current research focuses on virtual reality, augmented reality, human-computer interaction and visualization.

Zhiwen Shao is currently an Associate Professor with the China University of Mining and Technology, China, as well as a Postdoctoral Fellow with the Shanghai Jiao Tong University, China. He received the B.Eng. degree and the Ph.D. degree in Computer Science and Technology from the Northwestern Polytechnical University, China and the Shanghai Jiao Tong University, China in 2015 and 2020, respectively. From 2017 to 2018, he was a joint Ph.D. student at the Multimedia and Interactive Computing Lab, Nanyang Technological University, Singapore. He has published more than 40 academic papers in popular journals and conferences. His research interests lie in computer vision and affective computing. He has been serving as a Publication Chair for CGI 2023, as well as a PC member for IJCAI and AAAI.

Jieming Yin is a professor with the School of Computer Science, Nanjing University of Posts and Telecommunications, China. He obtained his PhD degree from University of Minnestate-of-the-art, Twin Cities in 2015. He used to be a faculty member at Lehigh University, and a researcher at AMD. His research interests lie in computer architecture, machine learning aided computer system design, and virtual reality.



Lili Wang received her Ph.D. degree from the Beihang University, Beijing, China. She is a professor with the School of Computer Science and Engineering of Beihang University, and a researcher with the State Key Laboratory of Virtual Reality Technology and Systems. Her interests include virtual reality, augmented reality, mixed reality, real-time rendering and realistic rendering.