International Journal of Human–Computer Interaction

ISSN: (Print) (Online) Journal homepage: https://www.tandfonline.com/loi/hihc20

Manipulation Guidance Field for Collaborative Object Manipulation in VR

Xiaolong Liu, Lili Wang & Shuai Luan

To cite this article: Xiaolong Liu, Lili Wang & Shuai Luan (29 Aug 2023): Manipulation Guidance Field for Collaborative Object Manipulation in VR, International Journal of Human–Computer Interaction, DOI: 10.1080/10447318.2023.2250941

To link to this article: https://doi.org/10.1080/10447318.2023.2250941

INTERNATIONAL JOURNAL OF HUMAN–COMPUTER INTERACTION
https://doi.org/10.1080/10447318.2023.2250941

Manipulation Guidance Field for Collaborative Object Manipulation in VR

Xiaolong Liu[a], Lili Wang[a,b], and Shuai Luan[a]

[a]State Key Laboratory of Virtual Reality Technology and Systems, Beihang University, Beijing, China; [b]Peng Cheng Laboratory, Shengzhen, China

ABSTRACT

Object manipulation is a fundamental interaction in virtual reality (VR). Efficient and accurate manipulation is important for many VR applications, especially collaborative VR applications. We introduce a collaborative method based on the manipulation guidance field (*MGF*) to improve manipulation accuracy and efficiency. *MGF* aims to guide users of different manipulation types to different manipulation viewpoints to efficiently and collaboratively manipulate objects. First, we introduce the concept of *MGF* and its construction method. Two strategies are offered to accelerate the *MGF* updating process. A collaborative manipulation method to manipulate objects using the guidance of MGF is then proposed. Finally, a user study ($n = 36$ participants) was conducted to evaluate the efficiency and accuracy of our *MGF*-based collaborative object manipulation method in three scenes: (1) Livingroom scene; (2) WaveHouse scene; (3) Pipe scene. Compared to a control method without *MGF*, the results show that our *MGF*-based method has significantly reduced task completion time, position error, rotation error, and task load.

KEYWORDS

Virtual reality; collaboration; object manipulation; guiding field; human-computer interaction

1. Introduction

Object manipulation is a fundamental interaction in product design, 3D object modeling and virtual object assembly in virtual reality (VR) applications. Object Manipulation most often refers to spatial transformation (Ruddle, 2005). There are several different types of spatial transformations: translation, rotation, scaling, shearing, and reflection, among others. But the most common transformations are translation and rotation, which are necessary for positioning tasks. Scaling has been combined with these two fundamental manipulations since the seminal work (Zeleznik et al., 1997). Together with selection, these three transformations have been identified as fundamental manipulation tasks that have been kept together in numerous other research efforts. The efficiency and accuracy of object manipulation directly impact usability and application performance. Object manipulation in VR includes single-user manipulation (Bowman and Hodges 1997; Gloumeau et al., 2020; Nguyen et al., 2014; Wang et al., 2011) and multi-user manipulation (Ruddle et al., 2002). Collaborative manipulation refers to the manipulation of the same object by multiple users, which enhances the team's ability to solve complex manipulation tasks and is essential for applications such as VR team equipment assembly and maintenance training (Grandi et al., 2019). And the collaborative strategy of multiple users in the process of manipulating objects is the key issue in collaborative manipulation.

For collaborative manipulation, some methods assign different manipulation types to different users. In the early work in this field, the user's location is fixed, and his/her manipulation type remained unchanged after pre-specification (Chenechal et al., 2016; Pinho et al., 2008). Pinho et al. (2008) proposed a method to allow users to translate the object only in one direction. Chenechal et al. (2016) assigned four users with different manipulation types. The first user has the global view and is responsible for translation; the second user of the internal view of the object is responsible for scaling and rotation. The third and fourth users contribute a third-person perspective. The third is responsible for scaling, and the fourth switches between the other users' viewpoints and helps them communicate verbally. Recently, the user has been allowed to move in the manipulation process. Lages (2016) proposed a method with a director to manually assign different manipulation to different users according to his/her observation. An alternative approach involves the user responsible for the rotation manipulation always follows the manipulated object, and the user responsible for the translation is in a fixed position in the distance (Soares et al., 2018).

The existing collaborative manipulation methods do not comprehensively analyze the manipulation viewpoints based on the scene and guide the user to select an appropriate manipulation viewpoint for manipulation during the manipulation process (Chenechal et al., 2016; Lages, 2016; Soares et al., 2018). For example, in VR design, a user may need to manipulate virtual objects within a 3D scene. However, current collaborative manipulation methods may not provide sufficient guidance on how to select an appropriate viewpoint for manipulating the object. For example, the user may be given a set of predefined viewpoints to

CONTACT Lili Wang wanglily@buaa.edu.cn; Xiaolong Liu liuxiaolong186@buaa.edu.cn State Key Laboratory of Virtual Reality Technology and Systems, Beihang University, Beijing, China

choose from, but these viewpoints may not be suitable for the specific task or scene. Without a comprehensive analysis of the manipulation viewpoints based on the scene, the user may waste time and effort selecting and trying out different viewpoints, which can be frustrating and lead to suboptimal results. In contrast, a more effective approach would be to analyze the scene and provide the user with guidance on which viewpoints would be most appropriate for manipulating specific objects or features within the scene. This would help streamline the manipulation process and improve the overall user experience in VR design. In some VR training applications, such as mechanical part assembly training and physical chemistry lab training, the target is known in these cases. The challenges of visual depth perception when using a VR system can indeed lead to limitations in accurately aiming and manipulating objects. Based on those, we introduce the concept of the manipulation guidance field (ie, *MGF*). To guide the user more reasonably and efficiently to the appropriate position for object manipulation, two problems need to be addressed. The first one is finding viewpoints suitable for a given manipulation type in the virtual scene when the object is manipulated. The second problem is guiding the user to the appropriate viewpoint to manipulate the object.

This paper introduces a collaborative method guided by the manipulation guidance field (*MGF*) to improve manipulation accuracy and efficiency in multi-user VR applications. *MGF* aims to guide users with different manipulation types to specific viewpoints that will allow them to manipulate objects efficiently and collaboratively. *MGF* is a discrete space vector field with each point associated with a vector ***M(T, R, S)***, representing the viewpoint quality for translation, rotation, and scale. We first give the concept of *MGF* and the construction method and propose two strategies to accelerate the *MGF* updating process. Constructing *MGF* requires a known target, and this information is available in many VR training applications. Then we propose a collaborative manipulation method using the *MGF*. Finally, we designed and conducted a user study to evaluate the efficiency and accuracy of our *MGF*-based collaborative object manipulation method. Prior to the commencement of the user study, we conducted a pilot user study to evaluate the effectiveness of *MGF*. Compared to the method without *MGF*, the results show that the *MGF*-based collaborative object manipulation method has significantly reduced task completion time, position error, rotation error, and task load. Figure 1 shows two users completing a task: Our *MGF*-based collaborative object manipulation method guides two users to manipulate the bunny to the target, which is highlighted by green. In Figure 1, the "T" value is used to guide the user for translating the object, and the visualization of the average of "R" and "S" is used to guide the user for rotating and scaling the object.

In summary, our main contributions are as follows:

- we introduce the concept of the manipulation guidance field and its construction method and propose two strategies to accelerate the *MGF* updating process;
- we propose a collaborative object manipulation method using the guidance of *MGF*;
- we design a user study to evaluate the efficiency and accuracy of our *MGF*-based collaborative object manipulation method. Compared to a control method without *MGF*, the results show that our *MGF*-based method has significantly reduced task completion time, position error, rotation error, and task load.

2. Related work

Efficient and accurate object manipulation is crucial for some virtual reality applications. Over the past 20 years, researchers have proposed many methods to improve the efficiency and accuracy of single-user object manipulation (Frees & Kessler, 2005; Gloumeau et al., 2020; Liu et al., 2022; Song et al., 2012; Wang et al., 2011; Wilkes & Bowman, 2008). Single-user object manipulation methods refer to techniques used by an individual to interact with and manipulate an object. For a detailed understanding of single-user object manipulation methods, readers are advised to read those survey papers (Bergström et al., 2021; Mendes et al., 2019). In this section, we review previous work on object manipulation by multi-users, viewpoint quality computation, and artificial potential field based guidance.

2.1. Collaborative object manipulation in VR

Collaborative manipulation refers to the process in which two or more people work together to manipulate an object. Compared to single-user manipulation, collaborative manipulation has several advantages: (1) Collaborative manipulation can assign different parts of a task to different individuals, allowing each person to be responsible for their expertise, which can improve the efficiency and accuracy of the manipulation task; (2) Collaborative manipulation can integrate different perspectives and ideas from a different user, thereby better understanding the manipulation task itself and improving the efficiency and accuracy of manipulation. (3) Collaborative manipulation reduces the likelihood of manipulation errors. Over the past two decades, many researchers have also studied collaborative manipulation. The ideas of collaborative manipulation methods are mainly divided into three classes. The first idea is that the manipulations of multiple users are integrated into different ways to manipulate objects. Ruddle et al. (2002) studied how to integrate collaborative manipulation. They compared two integrated action methods: one that selects the same part of different user manipulation, and the other calculates the average of different user manipulation. Duval et al. (2006) proposed a method for object manipulation by integrating each user's manipulation point and manipulation direction. The second idea is to improve the efficiency and accuracy of collaborative manipulation through visual feedback. Kai et al. (2006) proposed a bent-pick-ray method. When multiple users select the same object, the rays bend according to the pointing direction and the selected point, providing continuous visual feedback to the users. Baron (2016) proposed

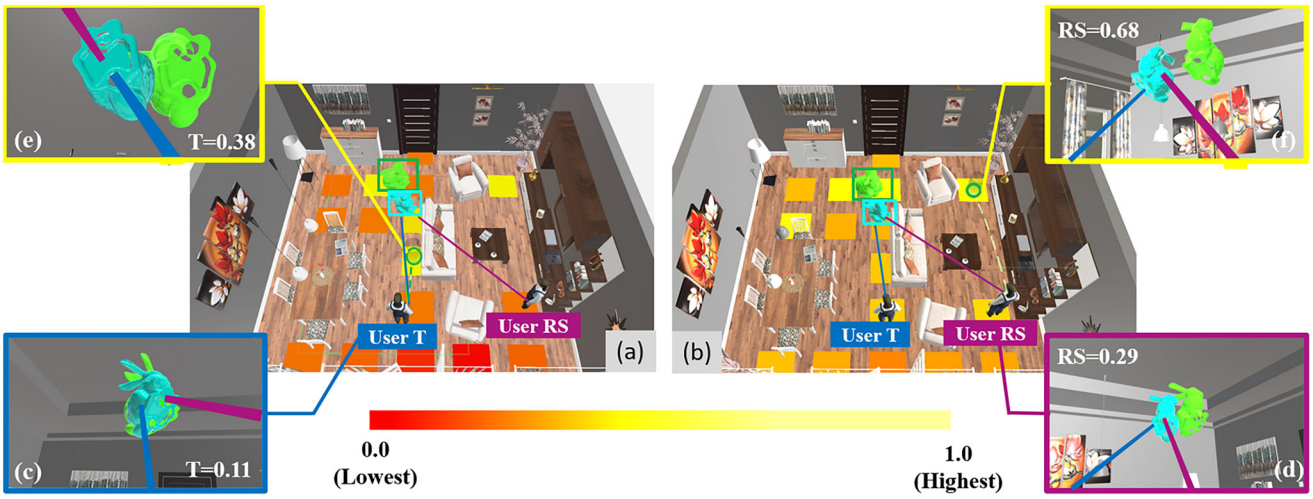

Figure 1. Two users collaboratively manipulate the bunny (cyan) to the target (green) position according to the cues of the manipulation guidance field. In (a) and (b), the colors of the squares on the floor visualize the values of the "T" and "RS" components of the manipulation guidance field. (c) and (e) show the user views when user *T* stands on the locations of the different squares (green circles indicate teleportation). Compared to (c), (e) has a higher *T* value and is the better location for translating the bunny to the target. (d) and (f) show the user views when user *RS* stands on the locations of the different squares. Compared to (d), (f) has a higher *RS* value and is the better location for rotating and scaling the bunny to the target.

a UI specification for collaborative manipulation, reducing communication requirements through synchronization mechanisms and visual feedback. Wang et al. (2021) proposed a method to automatically assign object manipulation dominator to different users through viewpoint quality. Manipulator dominator refers to users in VR multi-user collaboration who are particularly effective at manipulating virtual objects. This method only calculates the quality of multiple users' viewpoints without considering the quality of other viewpoints in the scene and cannot guide users to viewpoints with higher quality. The third idea is to assign different manipulation types according to the user's viewpoint to improve the efficiency and accuracy of collaborative manipulation.

There are two types of methods arising from this concept. One is that the user's viewpoint is fixed, and each user's manipulation type is fixed. Pinho et al. (2008) first proposed assigning different direction translation manipulation types to users according to the user's fixed viewpoint. Chenechal et al. (2016) assign different viewpoints to each user, in which the user of the global view is responsible for translation, and the user of the internal view of the object is responsible for scaling and rotation. There are also two users with a third-person perspective. One user is responsible for scaling, and the other switches between participants' viewpoints and helps them communicate verbally. Another is that the user's viewpoint changes. Lages (2016) proposed a method for the director to assign viewpoints and action types to other users in a virtual scene. Soares et al. (2018) assign different manipulation types to different users according to the distance between the user's initial position in the virtual scene and the manipulated object. The user responsible for rotation always follows the object, and the user responsible for translation is at a fixed position in the distance. In the existing collaboration manipulation methods, the relationship between the movement of the manipulated object, the target, and the scene is not fully considered.

2.2. Viewpoint quality computation

A good viewpoint for an object or scene can be defined as one that provides a clear and unobstructed view of the object or scene while highlighting important visual features. Additionally, a good viewpoint should allow the viewer to focus on the most relevant or interesting aspects of the object or scene, such as its shape, texture, or color, and provide a sense of depth and perspective. The concept of the general position of the viewpoint proposed by Kamada and Kawai (1988) refers to a specific location or angle from which an object can be viewed in a way that provides the maximum amount of shape information in the rendering image. Plemenos and Benayada (1996) proposed a new constraint for selecting a viewpoint that provides a good visualization of an object. According to their paper, a viewpoint that maximizes the angular deviation between the view direction and the surface normal of the object provides the best visualization of its geometric details. Vázquez et al. (2001) proposed a method for automatically exploring scenes based on viewpoint entropy. It takes into account both the projection area and the number of visible faces of the object. Sokolov and Plemenos (2005) proposed the concept of" viewpoint quality." They suggested that in the context of global world exploration, the quality of a viewpoint could be determined by two factors: Total curvature for meshes and the projection area of the visible region of the objects. And the method proposed by Sokolov et al. (2006) for calculating viewpoint quality in automatic 3D scene exploration is based on three factors: the size of the object bounding box, the observation quality, and the fraction of visible area of the object. Freitag et al. (2016) proposed a method to normalize the viewpoint quality values according to the viewpoint quality of the whole scene and used the normalized viewpoint quality to adjust the travel speed when traversing large scenes automatically. Then they (Freitag et al. 2018) proposed an interactive assist interface based on automatic analysis of object visibility and

viewpoint quality to support exploration and guide the user to the interesting parts of the scene. Key aspects of the viewpoint quality included the object's uniqueness and the visual size of the object. Wang et al. (2021) constructed a viewpoint quality function and evaluated the viewpoints of multiple users by calculating its three components: the visibility of the object needs to be manipulated, the visibility of the target, the depth and distance combined of the target.

The viewpoint quality computation has different calculation criteria for different tasks. For the task of collaborative object manipulation, different users are responsible for different manipulation tasks, so how to propose a viewpoint quality computation suitable for collaborative objects is key.

2.3. Artificial potential field base guidance

The artificial potential field approach is widely used in robotics, especially for mobile robot navigation, where the robot must navigate through an unknown environment while avoiding obstacles and reaching a target destination. Khatib (1985, 1987) proposed the concept of an artificial potential field, which is the sum of the obstacle-related avoidance vector and the target-related attraction vector. Dynamic trajectories that adapt to changing conditions can be continuously generated through artificial potential fields. Patil et al. (2011) proposed a new method to guide and control virtual crowds using navigation fields. This method solves the problem of directing the agent flow in the simulation and interactively controlling the simulation at run-time. In this method, the goal position of each agent can be calculated from a higher-level objective, and it can be dynamically changed during the simulation. Bachmann et al. (2019) introduced an artificial potential field in Redirected Walking (RDW) based on which the user is" pushed" away from obstacles and other users. However, the method did not take into account the reasonable turning targets of potential users in physical space. Therefore, Dong et al. (2020) proposed a new method of multi-user redirected walking using a dynamic artificial potential field. In addition to using repulsive force to keep users away from obstacles and other users, gravity is also used to guide users into open or unobstructed spaces. Messinger et al. (2019) proposed a revised version of the APF-RDW algorithm. The APF-RDW algorithm is designed to be adaptive to the shape of the tracking area and to effectively support multiple users in a shared virtual environment. In APF-RDW, each obstacle or boundary within the virtual environment is represented by a repulsive force that is proportional to the distance between the user and the obstacle or boundary. The main idea of our paper is to establish a real-time and context-specific manipulation guidance field for guiding multiple users in collaborative manipulation. This guidance field is constructed based on the scene, manipulated objects, and targets.

In order to more reasonably and effectively guide the user to an appropriate position for object manipulation, two problems need to be addressed. The first is finding viewpoints suitable for a given manipulation type in the virtual scene when the object is manipulated. The second problem is guiding the user to the appropriate viewpoint to manipulate the object. The target is known in these cases in some VR training applications, such as mechanical part assembly training and physical chemistry lab training. Based on this, we introduce the concept of the manipulation guidance field, its construction method and two strategies to accelerate the *MGF* update process. Additionally, we propose a collaborative manipulation method using the guidance of *MGF*. As far as we know, we propose the concept of the manipulation guidance field for the first time, and the construction of the manipulation guidance field is completely new.

3. Object manipulation guidance field

In this section, we first introduce the concept of the object manipulation guidance field in Subsection 3.1, then the *MGF* construction and updating methods are given in Subsection 3.2. An optimization method is provided to accelerate the updating in Subsection 3.3.

3.1. Definition

The users in different positions of the virtual scene are suitable for different manipulation types when they manipulate objects collaboratively. *MGF* aims to guide users of different manipulation types to different manipulation viewpoints to manipulate objects efficiently and collaboratively. *MGF* is a discrete space vector field, and each element corresponds to a viewpoint ***M(T, R, S)*** in the 3D space. Its three components, T, R, and S, represent the quality of the viewpoint for translation, rotation, and scale, respectively. The larger the value of "T"/"R"/"S," the more suitable the viewpoint is for translation/rotation/scale. We use Equation 1 to represent our *MGF*.

$$MGF = \boldsymbol{M}(T, R, S)_{m\times n} \tag{1}$$

3.2. Construction and updating

To construct and update the *MGF*, we consider the relation of the manipulated object and the target, as well as the occlusion generated by the scene. Before constructing and updating the *MGF*, the following three steps are required: (1) extract the walkable area of the scene; (2) generate viewpoints on the walkable area; (3) initialize the position and direction of the camera at each viewpoint. Then, we compute the values of $T/R/S$ of *MGF* according to the images rendered with the cameras.

Given a virtual scene, we first get the *Navimesh* of the virtual scene, on which the user can move freely without being hindered by environmental obstacles. A *Navimesh* is a collection of two-dimensional convex polygons (a polygon mesh) that define which areas of an environment are traversable by the user in VR. Based on *Navimesh*, we get the walkable area *WA* of the entire scene, as shown in Figure 2(b). We sample the locations inside *WA* uniformly with a horizontal interval of 1.5 m. At each location, two viewpoints are placed with heights of $1.0m$ (corresponding to the user's crouching posture) and 1.7 m (corresponding to the

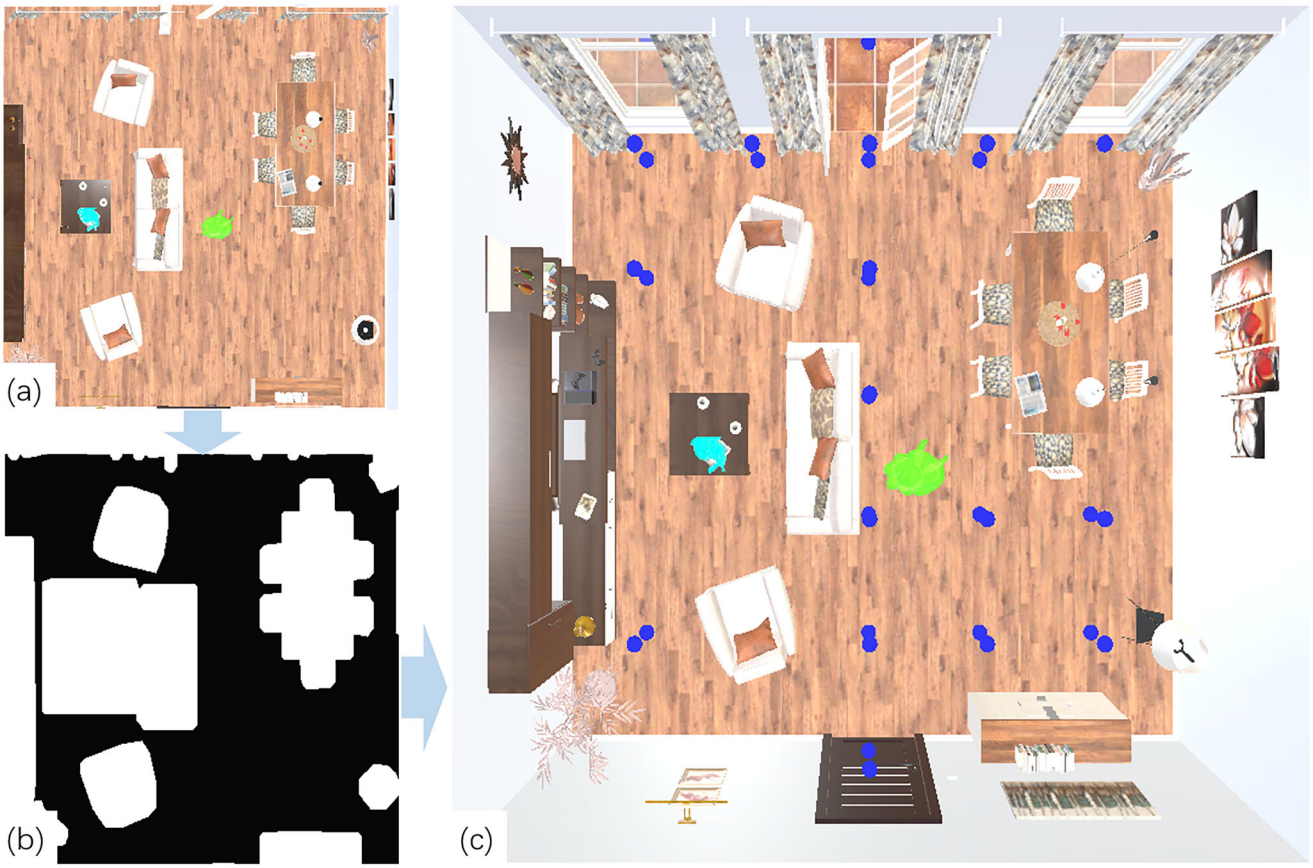

Figure 2. (a) is the top view of Scene. (b) is the walkable area *WA*. (c) is a third view of Scene, and the blue balls are the sampled viewpoints in (c).

person's normal standing posture). We call these sampled viewpoints *SV*, as shown in Figure 2(c). For each viewpoint, a camera is built towards the midpoint of the target and manipulated object. The FOV of the camera is 110°, as shown in Figure 3. We rendered an image from this camera for each output frame, with a 32×32 pixels resolution. We use this image to calculate the value of T/R/S.

The values of T, R, and S are updated for each output frame according to the object-target relation d, θ, p, and occlusions O_c in the image. Algorithm 1 is proposed to compute the object-target relation metric (d, θ, p), where d is the distance from the object to the target, θ is the angle difference between the object and the target, and p is the projected area ratio of the object and the target.

Algorithm 1. Object-target relationship metric

Input: object o, target t, viewpoint V, view I
Output: distance d, angle θ, area proportion p
1: b_o, b_t = OBB(V,o,t);
2: c_o, c_t =GetOBBCenter(b_o, b_t);
3: l_o, l_t = GetOBBAxis(b_o, b_t);
4: d = Distance(c_o, c_t);
5: θ = GetAngle(l_o, l_t);
6: A_o, A_t = Area(I, o, t);
7: $p = A_o/A_t$;
8: **return** (d, θ, p);

The inputs of this algorithm are the geometry of the manipulated object o and the target t, the current viewpoint V, and the image I rendered from V. The output is triple (d, θ, p). First, we project the object and target from V, and construct the oriented bounding box (OBB) (line 1). The centers and the long axes of the OBBs are obtained (lines 2–3). After this, the distance d and the angle difference θ between the object and the target are calculated (lines 4–5). we calculate the projection area of the manipulated object and the target and the ratio p of the projection area (lines 6–7). At last, the distance d, the angle difference θ, and the projected area ratio p are returned (Line 8).

We use Equation (2) to calculate the dis-occlusion factor O_c, where O_e is the scene occlusion area, which represents the total occlusion area of the scene to the target and the object; O_o is the occlusion area of the object to the target, S_o is the total area of the object, S_t is the total area of the target. Small values of O_e and O_o indicate that the two types of occlusion areas in the scene are small, so the overall dis-occlusion effect of the scene under the current viewpoint is good. Therefore, the larger the dis-occlusion factor O_c, the more suitable for user object manipulation.

$$O_c = \frac{\exp(-O_e/(1 + S_t + S_O))}{\exp(O_o/(1 + S_t))} \tag{2}$$

T, R, and S can be calculated with Equations (3)–(5). The larger the values of T, R, and S are, the more suitable the viewpoint is for the corresponding manipulation type of the user. In Equation (3), d is the distance in pixels between the object and the target in the current view, which we normalize with the height H and width W of the view. The

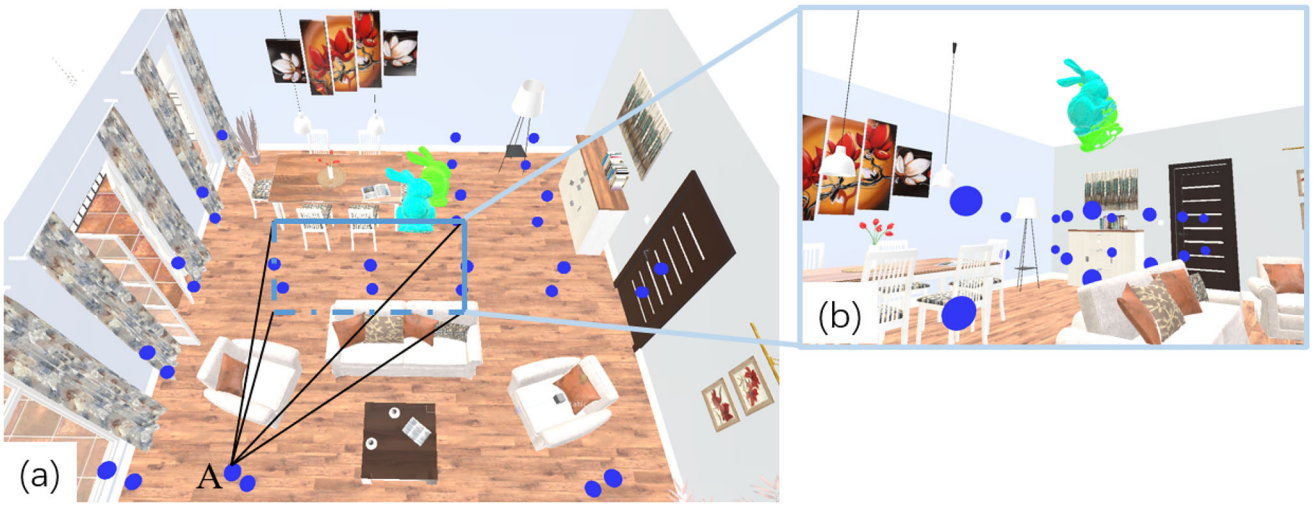

Figure 3. (a) is a third view of Scene. (b) is a view of viewpoint A in *SV*. The blue balls are the sampled viewpoints.

larger the value of $\frac{d}{\sqrt{W^2+H^2}}$, the more perpendicular the observation direction is to the line connecting the object and the target, and the closer the current viewpoint is to the object and the target. Therefore, the current viewpoint is more suitable for translation. In Equation (4), θ is the angle difference between the long axes of the bounding boxes of the object and the target in the current view. The view with the larger θ is more suitable for users to rotate the object. In Equation (5), p is the projected area ratio between the object and the target in the current view. The larger value of $(p-1)^2$, the larger the difference between the projected area of the object and the target, and the more suitable the viewpoint is for scale manipulation.

$$T = \alpha_t \frac{d}{\sqrt{W^2+H^2}} + \beta_t O_c \tag{3}$$

Where α_t is set to 0.7, β_t is set to 0.3, W is the width of the image V, H is the height of the image V, d comes from the triple (d, θ, p). The initial values (α_t, β_t) are set as 0.5. We use some images that are very easy to subjectively distinguish quality to test these weights and adjust the weights to make the results as reasonable as possible.

$$R = \alpha_r \frac{\theta}{\pi} + \beta_r O_c \tag{4}$$

Where α_r is set to 0.6 and β_r is set to 0.4, and θ comes from the triple (d, θ, p). The initial values (α_r, β_r) are set as 0.5. We use some images that are very easy to subjectively distinguish quality to test these weights and adjust the weights to make the results as reasonable as possible.

$$S = \alpha_s (p-1)^2 + \beta_s O_c \tag{5}$$

Where α_s is set to 0.4, β_s is set to 0.6, and p comes from the triple (d, θ, p). The initial values (α_s, β_s) are set as 0.5. We use some images that are very easy to subjectively distinguish quality to test these weights, and adjust the weights to make the results as reasonable as possible.

3.3. Optimization

When the scene is very large, the number of sampled viewpoints will be large. It is difficult to calculate in real-time if the value of *MGF* at all sampling viewpoints is updated simultaneously. Among the sampling viewpoints generated in *WA*, there are two types of viewpoints that do not need to update the *MGF*. The first type of viewpoint is blocked by other objects in the scene, and the objects and targets cannot be seen completely or even cannot be seen; the second type of viewpoint is far from the object and the target, making it difficult to observe the manipulated object and target. So we adopt two strategies to accelerate the *MGF* updating process.

3.3.1. Strategy 1. Reduce the number of sampled viewpoints

In the sampled viewpoints *SV* of the walkable area *WA*, some of the sampled viewpoints are far away from the manipulated object and the target, and the virtual environment occludes the manipulated object and the target in the image drawn from this viewpoint. These viewpoints are not conducive to the user's manipulation, so they need to be removed from *SV*. We define two ratios: (1) The ratio of the visible pixels r_{tar} of the target area in the viewpoint view to the total pixels of the image V; (2) The visible pixels of the manipulated object area in the r_{obj} viewpoint view to the total pixels of the image V.

In this strategy, we remove these inappropriate viewpoints according to r_{tar} and r_{obj}. When the distance between the target and the manipulated object is less than d_{Thres}, if $r_{tar} < Thres_1$, or $r_{obj} < Thres_3$, we delete the sampled viewpoints (Figure 4(a)).

When the distance between the target and the manipulated object is greater than d_{Thres}, the manipulation viewpoint camera will turn to face the target. If $r_{tar} < Thres_2$, we remove the sampled viewpoints (Figure 4(b)). In our implementation, we set d_{Thres} to $3\,m$, $Thres_1$ to 0.01, $Thres_2$ to 0.005, $Thres_3$ to 0.002. The final *MGF* viewpoints are shown in Figure 4(c).

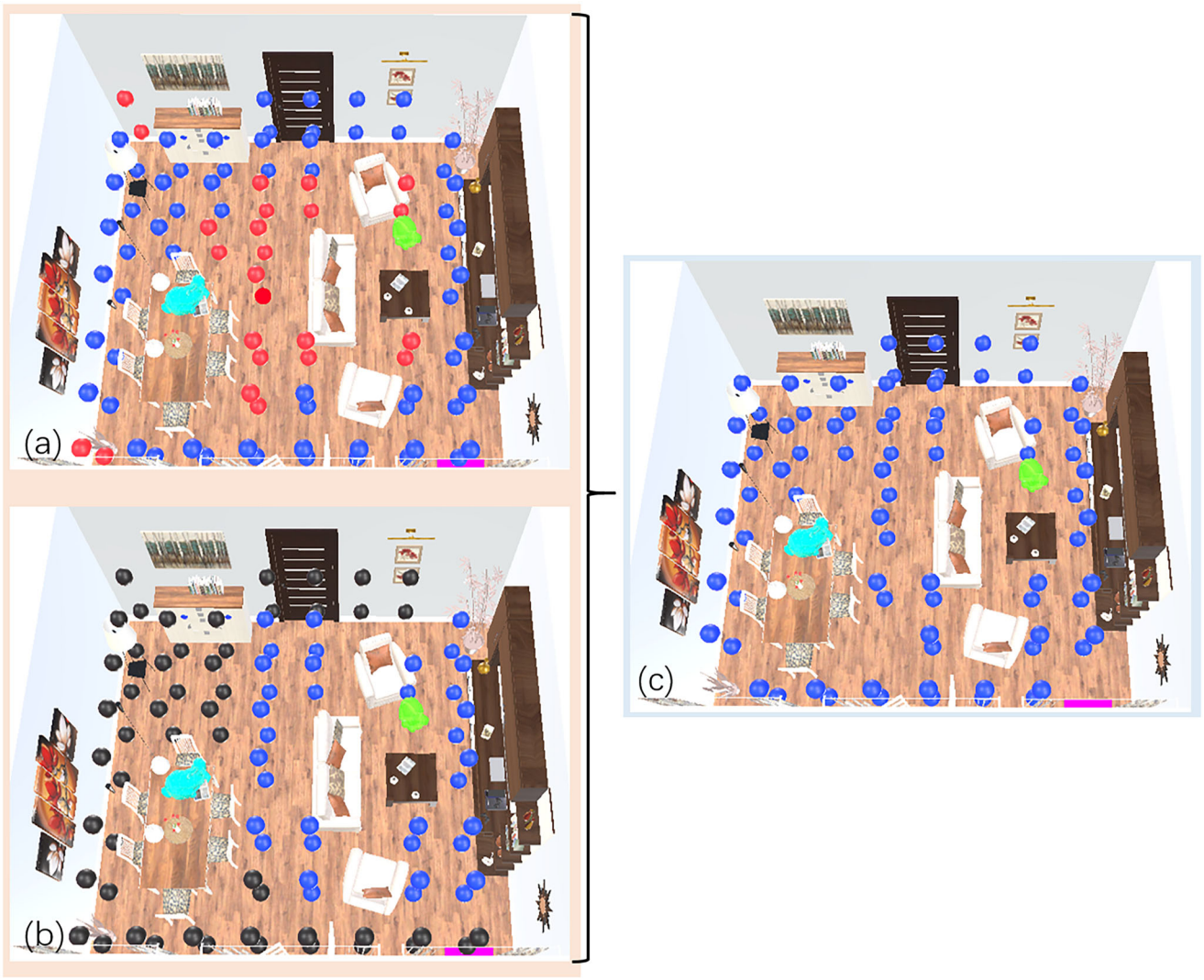

Figure 4. In (a), the red balls are the sampled viewpoints that are removed, and the blue balls are the remaining ones. In (b), the black balls are the sampling viewpoints that are deleted, and the blue balls are the retained balls. (c) is the final result of *MGF* viewpoints.

3.3.2. Strategy 2. Updating with a time interval

In order to reduce the time cost of the $\boldsymbol{M}(T, R, S)$ update, we reduce the update frequency. In a relatively short period, the relationship between the object and the target has not changed much. We divide the sampled viewpoints in *MGF* into four parts, which are updated every 0.2s.

4. Using MGF for guiding collaborative object manipulation

Our *MGF*-based collaborative object manipulation method provides viewpoint guidance for users with specific manipulation types by visualizing the values of T, R, and S. During our experimental testing, we asked the participants about their thought process while manipulating objects in virtual reality. All participants provided the same response: when translating an object, the user is concerned with the spatial location of the object and the target. However, when rotating and scaling objects, users pay more attention to the details and appearance of the objects. So the visualization of the "T" value is used to guide the user for translating the object, and the visualization of the average of "R" and "S" is used to guide the user for rotating and scaling the object. We use color changes to represent changes in the scores of viewpoints. We want to give feedback on three aspects: (1) guide users to a specific location; (2) provide users with an overview of the distribution of color squares in the entire scene, helping users decide to go to a specific location more easily;(3) guide the user to the exact viewpoint. So three feedback methods need to be provided. The color squares are placed on the floor to guide the user to a position where the user can better observe objects and targets. The mini-map gives users an overview of the distribution of color squares throughout the scene, helping users decide where to go more easily. The ball is placed in mid-air with two layers, and we want to guide the user's viewpoint to the exact viewpoint. So we design three ways to visualize *MGF*.

- **Color balls:** We visualize T or $(R + S)/2$ at all sampled viewpoints in *MGF* with small colored balls, as shown in Figure 5(a). The yellower the color, the larger the value

Figure 5. Viewpoints in the *MGF* are visualized as small spheres in (a). Viewpoints in the *MGF* are visualized as color squares in (b). A mini-map is added to give the global cues of *MGF* in (c).

of T or $(R + S)/2$, the redder the color, the smaller the value.

- **Color squares:** For each location with the sampled viewpoints, we render a colored square of size 0.5 m × 0.5 m on the floor. The color of the square is determined by the maximum value among the two sampled viewpoints in the vertical direction, as shown in Figure 5(b).
- **Mini-map:** This option adds a mini-map to give the user a cue of the global scene. The mini-map is the top view of the scenes with the color squares, and it also visualizes the user's current position and orientation with a blue triangle, as shown in Figure 5(c).

Two users collaboratively manipulate the object to the target position according to one of these three visualization ways. For the first two visualization methods, the color balls and color squares are always displayed during manipulation. For the third method, since placing the map directly in front of the user will hinder the user's observation of the 3D world, we place the abbreviated map icon in the upper right corner of the field of view, only when the user presses the "larger" button on the handle to request to view the map, the map is magnified and placed in the center of the user's field of view. According to the updating frequency of *MGF*, we update the visualization with 5 Hz. When the user translates, rotates, and scales the object, the visualization guides him to the viewpoint with a higher $T/R/S$ value (yellow location). Users can choose the natural walking or point-jumping (Bozgeyikli et al., 2016) method to go to the yellow location. After obtaining the *MGF*, we use different visualizations to guide the user to choose a viewpoint instead of directly switching to the best viewpoint. The reasons for this situation are: (1) The best viewpoint may be far away from the user's current location, and the user may not be able to see the best viewpoint at the user's location. If the user is placed directly at the best viewpoint, the continuity of observation is interrupted, possibly causing disorientation and sickness. (2) To maintain the continuity of observation, the user can be allowed to find the best viewpoint by himself. To find the best viewpoint, the user needs to observe all the viewpoints by repeatedly walking and jumping before making a decision, which requires additional time overhead. And the optimal position for rotation/scaling is not affected when the user translates the object. Two users can interact with the object simultaneously. When one user is responsible for rotation and scaling, the other user who is responsible for translation does not conflict with the user responsible for rotation and scaling.

5. User study

5.1. Pilot user study

We first designed a pilot user study to evaluate the effectiveness of *MGF*, ie the high-quality view calculated by *MGF* is largely consistent with the high-quality view selected by the user based on subjective feeling.

5.1.1. Participants

We recruited 12 participants through social platforms, 7 males, and 5 females, between 20 and 31 years old. Each participant spent 20–30 min, which rewarded 50 yuan. Seven of our participants had used HMD VR applications before. Participants had normal and corrected vision, and none reported vision or balance disorders. Two people form a group to manipulate the object collaboratively, one for translation and the other for rotating and scaling. Before the participants started the experiment, we first asked the participants to sign an informed consent form. Our pilot user study was awarded and approved by the Biology and Medical Ethics Committee of Beihang University.

5.1.2. Hardware and software setup

We used two sets of HTC Cosmos VR systems with two controllers, allowing two users to point virtual lasers at the virtual environment (VE). Each HMD is connected to its workstation with a 3.79 GHz Intel(R) Core(TM) i7-10700KF CPU, 16GB of RAM, and an NVIDIA GeForce GTX3090 graphics card. The tracked physical space hosting the VR applications is 4 m × 4 m. We used Unity 2019.1.9f1 to implement our VR collaborative manipulation tasks. The virtual environment is rendered at 90–100 fps per eye for user T and 60–90 fps per eye for user *RS*.

5.1.3. Manipulation implementation

We implemented the object manipulation method from (Wang et al., 2021) in our experiments. When the user keeps pressing the "on" button, the translation and rotation of the handle are 1:1 mapped to the manipulated object in

Figure 6. Image pair comparison in the pilot user study. Columns 1–3 are for the *Livingroom* scene, column 4 is for the *Wavehouse* scene and Column 5 is for the *Pipe* scene. The images with a green check mark are the better views users select at that moment, and the images in the green frames are the views with higher value evaluated with our method. The non-green frames are views with lower values evaluated using our method, and non-checkbox images are poorer views in the user-selected view. The images with a green check mark in columns 1–3 are the views that the user selected at the time to be more suitable for translating, and the images in the green frames are the views with higher values of the "T" component of the *MGF*. The images with green check marks in columns 4–5 are the views that the user selected that were more suitable for RS manipulation at that time, and the images in the green frames are the views with higher values of the "RS" component of *MGF*..

virtual space. Our method mainly focuses on the quality calculation and guidance of the observed viewpoints and a simple 1:1 mapping is used in the implementation. A better implementation of adaptive $1:N$ mapping (Frees et al., 2007) can be integrated into our method easily. The user can repeat the action to move the object away or close or reach a total rotation angle beyond the wrist limit, eg, keep pressing the "on" button, move, release the button, place the hand, and repeat. The user translates the object with his/her right hand and rotates the object with his/her left hand. The up and down directions of the joystick on the handle are used to scale the objects uniformly.

5.1.4. Task

The task of the pilot user study requires participants to choose the better viewpoint from the pair of viewpoints in each step of manipulating the object to the target position. Twelve people formed six groups: 1 group with the *Livingroom* scene, 2 for the *Wavehouse* scene, and 3 for the *Pipe* scene.

5.1.5. Procedure

We randomly set one participant ($participant_t$) to translate and one participant ($participant_{rs}$) to rotate and scale. Before each step of manipulation, the participants need to select two manipulation viewpoints in the scene for comparison by using teleportation. For example, $participant_t$ selects two viewpoints before each manipulation step, and then compares the two viewpoints, which viewpoint is more suitable for translation manipulation. The view after each teleportation is recorded. $participant_t$ needs to choose a manipulation viewpoint that is more suitable for translation, and $participant_{rs}$ needs to choose a manipulation viewpoint that is more suitable for rotation and scaling. After the two participants have determined the best manipulation viewpoint for the current step, the manipulation can be started by pressing the "OK" button, and the process can be repeated until the participants complete the manipulation.

5.1.6. Metric

We use metrics called $AccuracyRate_t$ and $AccuracyRate_{rs}$ to measure the effectiveness of *MGF*. After the users complete the manipulation tasks, we examine all view pairs generated throughout the process and their corresponding *T* and *RS*. We count the number of viewpoints n_{pt}, n_{prs} with higher *t*, *rs* values in the viewpoint pairs that are better viewpoints selected by the participants, and calculate the ratio of n_{pt}, n_{prs} to the number of pairs n_t, n_{rs} to get $AccuracyRate_t$, $AccuracyRate_{rs}$.

5.1.7. Results

In our pilot user study, n_{pt} is 163, n_t is 174. Therefore $AccuracyRate_t$ is 93.6%, which means that most people believe that the better views for translating the object at a certain moment are consistent with the views with higher scores calculated by using the guidance of *MGF*. n_{prs} is 157, n_{rs} is 187. Therefore, $AccuracyRate_{rs}$ is 83.7%, which means that most people subjectively believe that the better views for rotating and scaling the object at a certain moment are consistent with the higher scores calculated using the guidance of *MGF*. Figure 6 shows view pairs for the user's first selection (line 1) and second selection (line 2). Images marked with a check mark are the better views that the user selected at the time, and images marked with green frames are the views with higher scores calculated using the evaluation function in our *MGF*. By comparing the inconsistent view pairs, we find that the possible reason for the inconsistency is that we do not consider the geometric features of the object occlusion by the environment and objects. In conclusion, *MGF* is effective in most cases. That is the high-quality viewpoint image calculated by *MGF* is largely consistent with the high-quality view selected by the user based on subjective feelings.

5.2. User study design

We designed a user study with a manipulation task in 3 scenes to evaluate our method's efficiency, accuracy, and task load. The hardware settings and manipulation implementation used in the user study are the same as Subsection 5.1.

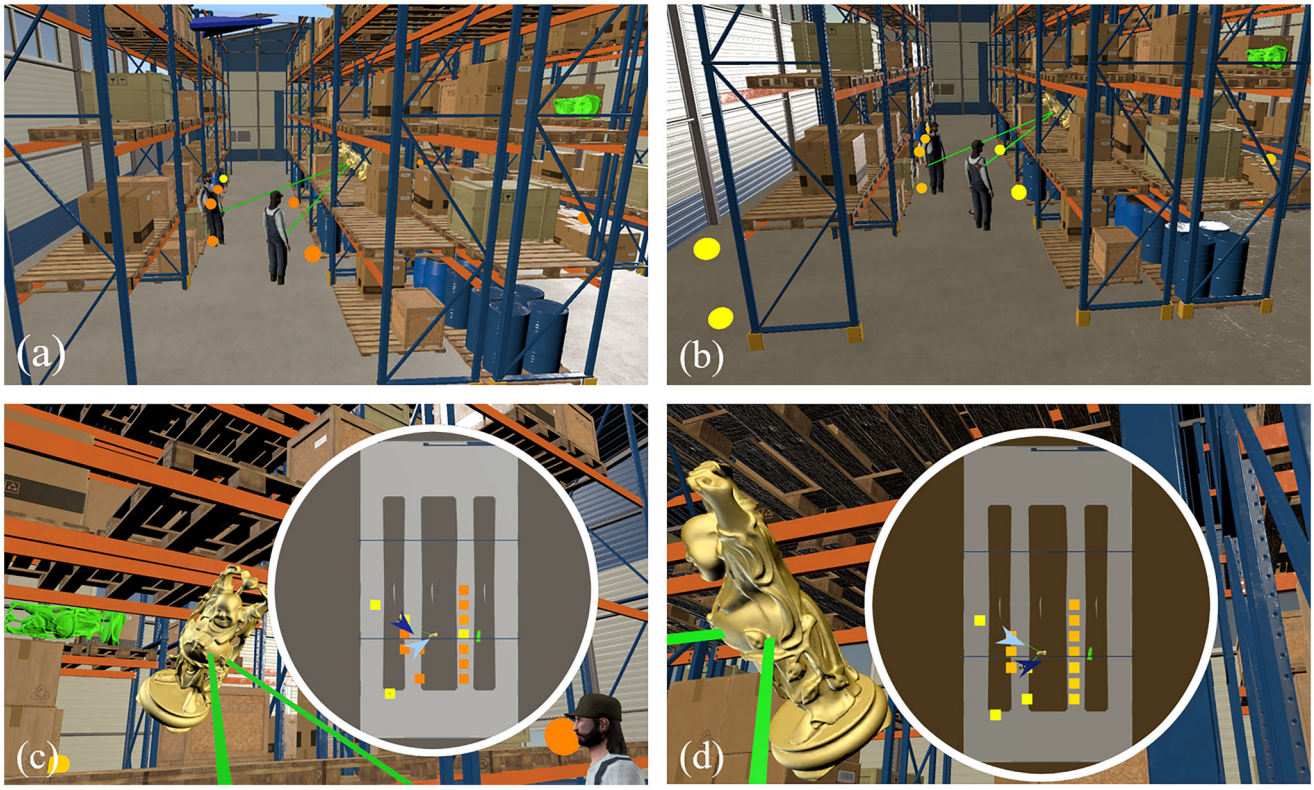

Figure 7. The second scene *S2* of our user study. (a) and (b) Show two participants manipulating Maitreya to a green target position guided by their respective small balls with a map of *MGF*. (c) and (d) show views seen from two viewpoints of two participants.

5.2.1. Participants

We have recruited 36 participants through social platforms, 30 males, and 6 females, between 20 and 31 years old. 24 of our participants had VR experience before. Each participant spent 50–60 min on every scene, which rewarded 100 yuan. If not all participants have used HMD VR applications, the findings of a user study may be limited in their generalizability and may not accurately reflect the experiences of individuals who have not had any experience with VR technology. However, many users have not used HMD VR applications at present. To reduce the influence of participants who have not used HMD VR applications on the experiment, let users fully understand the HMD VR application before the formal experiment and then conduct the experiment. Participants had normal and corrected vision, and none reported vision or balance disorders. Participants in the pilot study (Subsection 5.1) did not participate in this user study. There are one control condition and five experimental conditions. Condition *CC* is for an intuitive method without *MGF*, in which participants were not prompted for any information and chose to manipulate the viewpoint by themselves. Experimental conditions EC_1 to EC_5 use our method with different *MGF* visualization. EC_1 is with the small balls, EC_2 is with the mini-map, EC_3 is with the color squares, EC_4 is with the small balls and the mini-map, and EC_5 is with the color squares and the mini-map. The mini-map of EC_2, EC_4, and EC_5 is placed in the upper right corner of the user's view. Participants can magnify the map in the center of view through the "larger" button on the handle. The EC2 method, a commonly used method in VR (Zagata et al., 2021), was added to our study. However, our initial experiment found that users were slow when using EC_2 alone without combining it with other feedback methods. Users tended to spend a long time observing the small map but had difficulty quickly establishing the correspondence between the map and the VE. As a result, the time spent using the small map with EC_2 was approximately 2.5 times longer than that of EC_4 and EC_5. Therefore, we proposed the combinations of the mini-map with two other feedback methods (EC_4 and EC_5) to improve the manipulation accuracy and efficiency in VR.

5.2.2. Hypotheses

Our method was designed to allow the user to manipulate an object to the target efficiently. Thus, we formulate the following hypotheses:

For efficiency, we formulate the following hypotheses:

H0: The time it takes for users to manipulate the object to the target using EC_1 and EC_{3-5} is close compared to *CC*.

H1: Users can manipulate an object to the target faster with EC_1 and EC_{3-5} compared to *CC*.

For accuracy, we formulate the following hypotheses:

H0: Users can manipulate objects to target with EC_1 and EC_{3-5} with similar accuracy compared to *CC*.

H2: Users can manipulate an object to the target more accurately with EC_1 and EC_{3-5} compared to *CC*.

For task load, we formulate the following hypotheses:

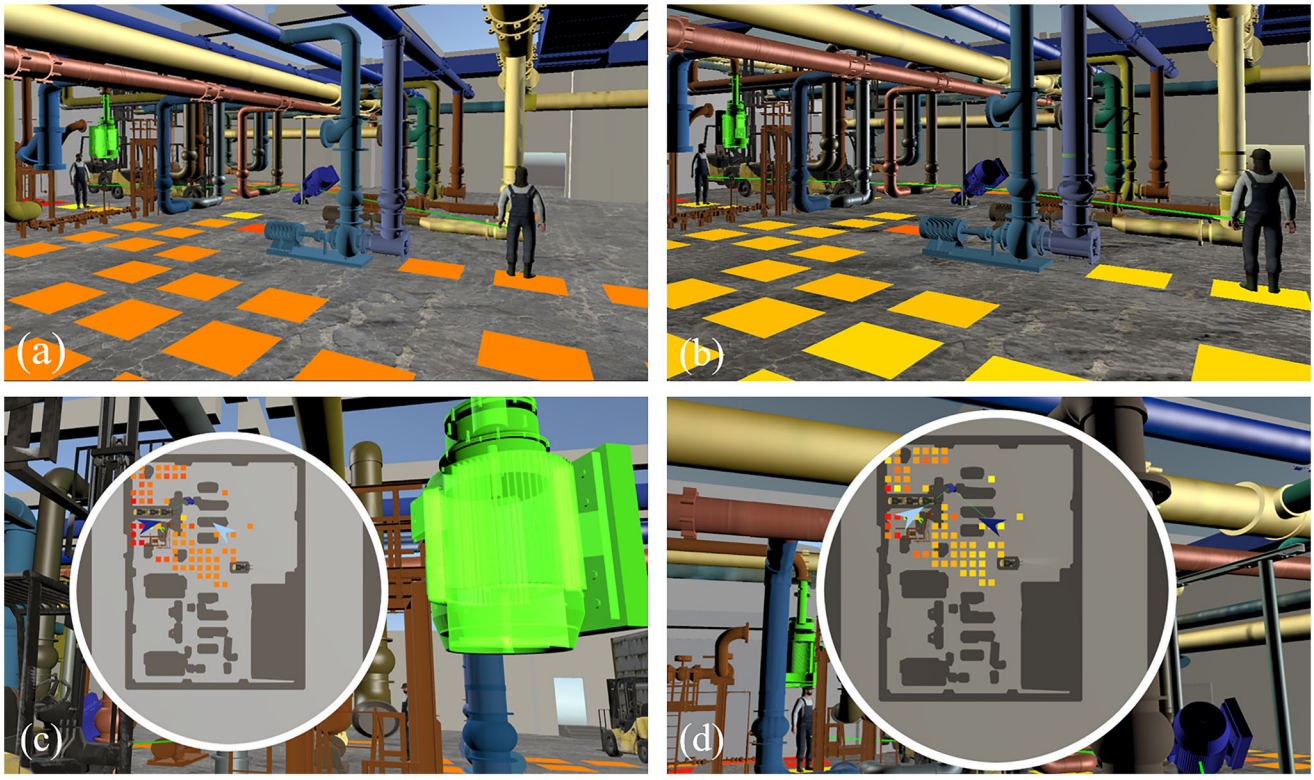

Figure 8. The third scene *S3* of our user study. (a) and (b) Show two participants manipulating blue pipe to a green target position guided by their respective color squares with map of *MGF*. (c) and (d) show views seen from two viewpoints of two participants.

H0: EC_1 and EC_{3-5} have the same task load as *CC*.
H3: Task load of EC_1 and EC_{3-5} is lower than that of *CC*.

5.2.3. Task

During the task, the users are required to manipulate the object as quickly and accurately as possible to a predefined target position. There are 3 scenes in the task. The target in each scene is fixed, and the user's position is placed at random scene locations in the initialization. After the two users collaborate to manipulate the object to the target, press the "end" button to complete the task.

In the *Livingroom* scene (*S1*), the participants are required to manipulate a bunny on the table to the target. The size of the *Livingroom* scene is 8.4 m × 9.2 m (Figure 1), and the size of the target cube is 0.8 m × 0.79 m × 0.62 m (Scale (8)), the rotation angle is 137.34°. The number of manipulated viewpoints for *MGF* of *S1* is 35. In the *Warehouse* scene (*S2*), the users are required to manipulate the *Maitreya* to the target. The size of the *Warehouse* scene is 16 m × 30 m (Figure 7), and the target size is about 0.47 m × 1.16 m × 0.47 m (Scale (5.8)), the rotation angle is 116.67°. The number of manipulated viewpoints for *MGF* of *S2* is 21. In the *Pipe* Scene (*S3*), the participants are required to manipulate a piece of blue component to the target position (Figure 8). There are many occlusions in the scene. The size of the *Pipe* scene is 50 m × 60 m, and the target size is about 2.4 m × 1.1 m × 0.8 m (Scale (3)), the rotation angle is 110.43°. The number of manipulated viewpoints for *MGF* of *S3* is 124.

5.2.4. Procedure

For *CC*, EC_1, EC_2, EC_3, EC_4, and EC_5, two participants form a group for collaborative manipulation. In the three scenarios, all groups performed the co-manipulation tasks with all conditions in Latin square random order. The minimum interval between the scenarios is one day, and the maximum interval is three days. The user study lasted about 15 d. Before the participants put on the VR headset, we first asked the participants to sign an informed consent form. The user study was awarded and approved by the Biology and Medical Ethics Committee of Beihang University. Before starting the experiment in each scene, two participants need to select at least three manipulation viewpoints and see the visual effect that the manipulated object and the target completely match. Then participants practice for 1 min before the task starts. When both users point to the object that needs to be manipulated, our system starts recording time and other objective metrics. We tell the participants that we will record and evaluate the task completion time, which indirectly encourages them to complete the task as soon as possible. We determined whether the participants were left-handed before the experiments, and our participants both were right-handed.

5.2.5. Metrics

Task performance was measured with the following objective metrics: (1) task completion time, in seconds, represents the time from when both collaborators point to the object until they both press the "end" button to confirm the end of the manipulation; (2) position error, in millimeters, indicates

the distance from the center of the manipulated object to the center of the target position when the participant presses the "end" button; (3) rotation error, in degrees, indicates the angle difference between the local coordinate system of the manipulated object and the target coordinate system when the participant presses the "end" button. If the angle difference of the three coordinate axes is α, β, γ, the rotation error is $\sqrt{\alpha^2+\beta^2+\gamma^2}$; (4) scale error, in times, indicates the ratio of the absolute value of the difference between the diagonal length of the bounding box of the manipulated object and the diagonal length of the target bounding box to the diagonal length of the target bounding box when the participant presses the "end" button; (5) teleportation number indicates the number of teleportation from both collaborators point to the object until they both press the "end" button to confirm the "end" of the manipulation. We also evaluated the perception with one subjective metrics: user task load, measured with the standard NASA TLX questionnaire (Hart, 2006; Hart & Staveland, 1988). After each condition in the session, the data of the task-load questionnaire are collected. We rank all the methods, and the higher the ranking, the higher the score. The calculation method of its rank score is as follows: $Score = (\sum_{k=1}^{n} Frequency \times Weight)/12$. The weight is determined by where the options are arranged. For our rank score: there are 6 options to participate in the sorting, the weight of the first position is 6, the weight of the second position is 5, the weight of the third position is 4, and the weight of the fourth position is 3, the weight of the fifth position is 2, and the weight of the sixth position is 1. For example, if the questionnaire is filled 12 times, EC_3 ranks in the first position 2 times, the second position 4 times, and the third position 6 times, then the average comprehensive score of EC_3: $Score = (2\times 6+4\times 5+6\times 4)/12 = 4.66$.

5.2.6. Statistical analysis

For each metric, the values of *CC* were compared to those of EC_1, EC_2, EC_3, EC_4, and EC_5, respectively, using a two-way repeated-measures ANOVA. First, the distribution normality assumption was verified using the Shapiro-Wilk test (Shaphiro & Wilk, 1965). Then the sphericity assumption is evaluated using the Mauchly test (Mauchly, 1940). A Greenhouse-Geisser correction is applied to the data when the sphericity assumption is violated. Then an overall ANOVA was conducted to investigate whether one can reject the null hypothesis that there is no statistically significant difference between the five conditions. When the null hypothesis was rejected ($p < 0.05$), the differences between the four pairs were analyzed with post-hoc tests, with a significance level lowered conservatively using the Bonferroni correction. For the time-dependent variable, we also quantified the size of the effect using Cohen's *d* (Cohen, 2013). The statistical analysis was performed using the SPSS software (IBM, n.d.). The *d* values were translated to qualitative effect size estimates of *Huge* ($d > 2.0$), *Very Large* ($2.0 > d > 1.2$), *Large* ($1.2 > d > 0.8$), *Medium* ($0.8 > d > 0.5$), *Small* ($0.5 > d > 0.2$), and *Very Small* ($0.2 > d > 0.01$) (Cohen, 2013).

Table 1. The completion time, in seconds.

Task	Condition	Avg ± std. dev.	$(CC_i\text{-}EC)/CC_i$	p	Cohen's *d*	Effect size
S1	*CC*	144.25 ± 47.36				
	EC_1	84.78 ± 16.82	70.2%	< 0.001*	1.67	Very large
	EC_2	130.90 ± 31.12	10.2%	0.311	0.33	Small
	EC_3	72.23 ± 22.16	99.7%	< 0.001*	1.95	Very large
	EC_4	88.70 ± 22.16	62.6%	< 0.001*	1.48	Very large
	EC_5	74.05 ± 17.54	94.8%	< 0.001*	1.97	Very large
S2	*CC*	114.90 ± 32.51				
	EC_1	95.30 ± 24.49	20.6%	0.042*	0.68	Medium
	EC_2	102.90 ± 23.77	11.7%	0.202	0.42	Small
	EC_3	84.75 ± 24.82	35.6%	0.002*	1.04	Large
	EC_4	89.30 ± 36.13	28.7%	0.027*	0.74	Medium
	EC_5	74.85 ± 27.47	53.5%	< 0.001*	1.33	Very large
S3	*CC*	135.75 ± 50.75				
	EC_1	106.65 ± 29.51	27.3%	0.037*	0.70	Medium
	EC_2	154.70 ± 42.13	−12.2%	0.218	0.41	Small
	EC_3	93.70 ± 23.30	44.9%	0.002*	1.06	Large
	EC_4	98.15 ± 28.36	38.3%	0.007*	0.91	Large
	EC_5	86.50 ± 22.66	56.9%	0.007*	1.25	Very large

Note. The (*) indicate that the result is statistically significant at $p < 0.05$.

5.3. Results

The results for evaluating task performance (Section 5.3.1) and perception (Section 5.3.2) are reported and discussed in the following subsections.

5.3.1. Task performance

5.3.1.1. Task completion time. Table 1 gives the task completion time. Statistical significance is indicated by an asterisk. The sphericity assumption is violated: $p < 0.001(S1), p < 0.001(S3)$. After applying the Greenhouse-Geisser correction, the overall ANOVA reveals significant differences between the five conditions: $(F_{2.374,66.469} = 45.801, p < 0.001)$ for S1, $(F_{5,135} = 8.378, p < 0.001)$ for S3. The sphericity assumption is verified: $p = 0.264(S2)$. The overall ANOVA reveals that there is a statistically significant difference between those conditions for *S2* $((F_{3.245,87.697} = 19.475, p < 0.001))$. Post-hoc analysis reveals that *CC* was significantly longer than for EC_1, EC_3,EC_4 and EC_5 for all three scenes. Compared with control conditions *CC* of all three scenes, EC_1, EC_3, EC_4 and EC_5 significantly improve the task time performance, and the effect size ranges from "Medium" to "Very Large," EC_2 does not significantly improve the task time performance. And We have put the results of the other pairwise comparisons at the end of the manuscript in the form of supplementary. From the results of the variance analysis of two-factor design, we can see that the main effect of scenes is not significant ($F = 1.117$, $p \le 0.160$, $\eta^2 = 0.006$); the main effect of feedback method is significant ($F = 47.11$, $p < 0.001$, $\eta^2 = 0.324$), main effect exists; the interaction effect between scene and feedback method is significant ($F = 3.531$, $p < 0.001$, $\eta^2 = 0.067$), interaction effect exists. Simple effects Shown: Under the *CC* method, the simple effect of the scene is significant ($F = 8.051$, $p < 0.001$, $\eta^2 = 0.032$); under the EC_1 method, the simple effect of the scene is significant ($F = 3.164$, $p = 0.031$, $\eta^2 = 0.013$); Under the EC_2 method, there is no significant difference in the simple effect of the scenario ($F = 2.186$, $p = 0.061$, $\eta^2 = 0.011$); Under the EC_3 method, the simplicity effect of the scenario is significant ($F = 22.236$, $p < 0.001$, $\eta^2 = 0.083$); Under the EC_3 method, there is no significant difference in the simple effect of the

Table 2. The position error, in millimeters.

Task	Condition	Avg ± std. dev.	$(CC_i\text{-}EC)/CC_i$	p	Cohen's *d*	Effect size
S1	*CC*	5.4 ± 2.0				
	EC_1	3.0 ± 1.4	80.4%	0.0016*	1.36	Very large
	EC_2	4.5 ± 1.9	19.7%	0.1678	0.46	Small
	EC_3	3.7 ± 1.2	47.2%	0.0027*	1.04	Large
	EC_4	3.2 ± 0.7	67.5%	< 0.001*	1.45	Very large
	EC_5	3.1 ± 0.7	75.9%	0.0016*	1.36	Very large
S2	*CC*	3.8 ± 2.2				
	EC_1	2.2 ± 0.9	74.6%	0.0035*	0.98	Large
	EC_2	3.8 ± 2.7	1.3%	0.952	0.02	Very small
	EC_3	2.5 ± 1.4	52.6%	0.0310*	0.73	Medium
	EC_4	2.5 ± 1.3	49.6%	0.0367*	0.70	Medium
	EC_5	2.1 ± 1.1	82.9%	0.0036*	1.01	Large
S3	*CC*	3.6 ± 1.2				
	EC_1	2.9 ± 1.5	22.8%	0.141	0.49	Small
	EC_2	3.2 ± 1.7	11.3%	0.460	0.24	Small
	EC_3	2.2 ± 1.0	62.6%	0.0006*	1.21	Very Large
	EC_4	3.1 ± 1.5	26.9%	0.0714	0.60	Medium
	EC_5	2.3 ± 1.0	57.9%	0.0007*	1.19	Large

Note. The (*) indicate that the result is statistically significant at $p < 0.05$.

Table 3. The rotation error, in degrees.

Task	Condition	Avg ± std. dev.	$(CC_i\text{-}EC)/CC_i$	p	Cohen's *d*	Effect size
S1	*CC*	5.52 ± 3.48				
	EC_1	3.62 ± 0.67	52.6%	0.0246*	0.76	Large
	EC_2	4.19 ± 0.97	31.9%	0.115	0.52	Medium
	EC_3	3.22 ± 1.42	71.4%	0.0111*	0.87	Large
	EC_4	3.29 ± 1.11	68.0%	0.0112*	0.86	Large
	EC_5	3.15 ± 1.34	75.3%	0.008*	0.90	Large
S2	*CC*	6.12 ± 2.56				
	EC_1	3.79 ± 0.75	61.6%	0.0004*	1.24	Very large
	EC_2	3.84 ± 0.90	59.2%	0.070	0.39	Small
	EC_3	3.18 ± 1.11	92.3%	< 0.0001*	1.49	Very large
	EC_4	3.69 ± 1.48	66.0%	0.0009*	1.16	Large
	EC_5	3.02 ± 1.25	102.8%	< 0.0001*	1.54	Very large
S3	*CC*	4.20 ± 1.92				
	EC_1	2.50 ± 0.97	68.4%	0.0013*	1.12	Large
	EC_2	3.26 ± 0.98	29.4%	0.062	0.62	Medium
	EC_3	2.32 ± 1.14	80.8%	0.0007*	1.19	Large
	EC_4	2.54 ± 1.39	80.0%	0.0039*	0.86	Large
	EC_5	2.12 ± 1.30	97.90%	0.0003*	1.29	Very large

Note. The (*) indicate that the result is statistically significant at $p < 0.05$.

scenario ($F = 2.186$, $p = 0.061$, $\eta^2 = 0.011$); Under the EC_4 method, there is no significant difference in the simple effect of the scene ($F = 0.759$, $p = 0.469$, $\eta^2 = 0.003$); Under the EC_5 method, there is no significant difference in the simple effect of the scenario ($F = 0.828$, $p = 0.438$, $\eta^2 = 0.003$).

5.3.1.2. Position error. Table 2 shows the position errors all conditions for these three scenes. The sphericity assumption is violated: $p < 0.001(S1, S2), p = 0.010(S3)$. After applying the Greenhouse-Geisser correction, the overall ANOVA reveals significant differences between the five conditions: ($F_{3.006, 69.139} = 10.053, p < 0.001$) for S1, ($F_{3.196, 86.288} = 6.067$, $p = 0.001$) for S2, and ($F_{3.352, 90.514} = 3.044, p = 0.012$) for S3. Post-hoc analysis reveals that *CC* was significantly larger than for EC_3 and EC_5 for all three scenes. Compared with control conditions *CC* of all three scenes, EC_3, EC_5 reduced position error significantly, and the effect size ranges from "Medium" to "Very Large"; And We have put the results of the other pairwise comparisons at the end of the manuscript in the form of supplementary. From the results of variance analysis of the two-factor design, we can see that the main effect of scenes is significant ($F = 1.714$, $p = 0.916$, $\eta^2 = 0.007$), the main effect does not exist; the main effect of feedback method is significant ($F = 14.922$, $p < 0.001$, $\eta^2 = 0.139$), the main effect exists; the interaction effect between scene and feedback method is not significant ($F = 1.52$, $p = 0.129$, $\eta^2 = 0.032$), the interaction effect is very small.

5.3.1.3. Rotation error. Table 3 gives the rotation error of all conditions for these three scenes. The sphericity assumption is violated: $p < 0.001(S1, S2), p = 0.003(S3)$. After applying the Greenhouse-Geisser correction, the overall ANOVA reveals significant differences between the five conditions: ($F_{2.078, 47.799} = 6.357, p = 0.003$) for S1, ($F_{2.529, 68.270} = 14.578, p < 0.001$) for S2, and ($F_{3.253, 87.827} = 10.632, p < 0.001$) for S3. Post-hoc analysis reveals that EC_1, EC_3, EC_4 and EC_5 were reduced rotation error significantly than for *CC* for all three scenes. Compared with control conditions *CC* of all three scenes, EC_1, EC_3, EC_4 and EC_5 significantly improves the task time performance, and the effect size ranges from "Medium" to "Very Large", EC_2 does not reduced rotation error significantly. And We have put the results of the other pairwise comparisons at the end of the manuscript in the form of supplementary. From the results of variance analysis of two-factor design, we can see that the main effect of scenes is not significant ($F = 1.784$, $p = 0.169$, $\eta^2 = 0.008$), the main effect does not exist; the main effect of feedback method is significant ($F = 29.214$, $p < 0.001$, $\eta^2 = 0.240$), the main effect exists; the interaction effect between scene and feedback method is not significant ($F = 0.256$, $p = 0.988$, $\eta^2 = 0.006$), the interaction effect is very small.

5.3.1.4. Scale error. Table 4 shows the scale errors of all conditions for these three scenes. The sphericity assumption is violated: $p < 0.001(S1, S2, S3)$. After applying the Greenhouse-Geisser correction, the overall ANOVA reveals not significant differences between the five conditions: ($F_{2.439, 60.968} = 2.722, p = 0.063$) for S1, ($F_{1.824, 43.775} = 0.520, p = 0.761$) for S2, and ($F_{3.251, 87.765} = 0.664, p < 0.651$) for S3. Post-hoc analysis reveals that EC_{1-5} were not significantly smaller than for *CC* for all scenes. From the results of variance analysis of the two-factor design, we can see that the main effect of scenes is not significant ($F = 1.127$, $p = 0.176$, $\eta^2 = 0.006$), the main effect does not exist; the main effect of the feedback method is not significant ($F = 0.76$, $p = 0.671$, $\eta^2 = 0.006$), the main effect does not exist; the interaction effect between scene and feedback method is not significant ($F = 0.231$, $p = 0.003$, $\eta^2 = 0.067$), the interaction effect is very small.

5.3.1.5. Teleportation number. Figure 9 shows the teleportation means of all conditions for these three scenes. The sphericity assumption is violated: $p = 0.03(T), p < 0.001(RS)$. After applying the Greenhouse-Geisser correction, the overall ANOVA reveals significant differences between the five conditions: ($F_{3.701, 118.423} = 4.991, p = 0.001$) for T, ($F_{2.406, 76.985} = 3.443, p = 0.029$) for RS. Post-hoc analysis reveals that the T of EC_{1-5} were not significantly larger than for *CC* for all scenes. The RS of EC_{1-5} were not significantly smaller than for *CC* for all scenes.

Table 4. The scale error, in times.

Task	Condition	Avg ± std. dev.	$(CC_i\text{-}EC)/\ CC_i$	p	Cohen's *d*	Effect size
S1	*CC*	0.020 ± 0.010				
	EC_1	0.014 ± 0.013	49.9%	0.275	0.73	Medium
	EC_2	0.020 ± 0.013	0.5%	0.982	0.01	Very small
	EC_3	0.013 ± 0.012	54.9%	0.138	0.63	Medium
	EC_4	0.013 ± 0.007	54.9%	0.090	0.63	Medium
	EC_5	0.011 ± 0.007	70.4%	0.090	0.74	Medium
S2	*CC*	0.033 ± 0.018				
	EC_1	0.030 ± 0.018	8.9%	0.654	0.15	Very small
	EC_2	0.029 ± 0.010	13.8%	0.410	0.27	Very small
	EC_3	0.032 ± 0.022	1.5%	0.948	0.02	Very small
	EC_4	0.025 ± 0.006	28.7%	0.104	0.63	Medium
	EC_5	0.030 ± 0.022	7.2%	0.743	0.11	Very small
S3	*CC*	0.012 ± 0.007				
	EC_1	0.011 ± 0.007	2.9%	0.886	0.10	Very small
	EC_2	0.012 ± 0.007	0.1%	0.956	0.01	Very small
	EC_3	0.010 ± 0.005	18.6%	0.441	0.29	Small
	EC_4	0.011 ± 0.004	2.9%	0.805	0.05	Very small
	EC_5	0.011 ± 0.010	7.7%	0.769	0.10	Very small

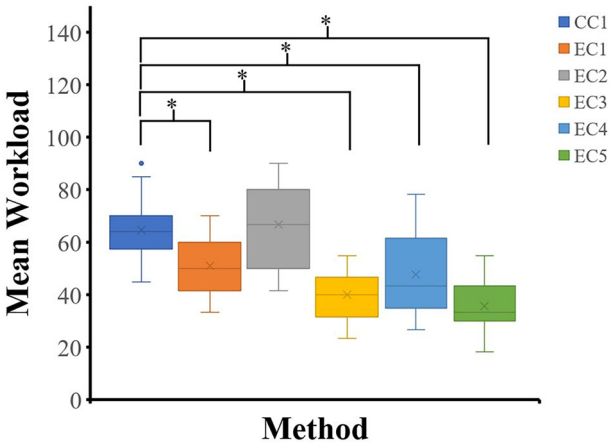

Figure 10. Box plots for task load scores of the six conditions in the all scenes. Asterisks denote statistical significance.

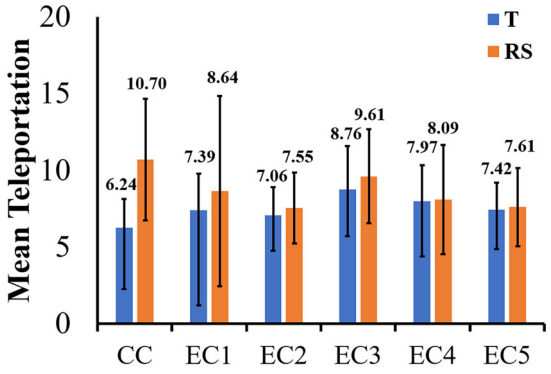

Figure 9. Mean teleportation. *T* is the teleportation number of the user *T*; RS is the teleportation number of the user RS. Error bars indicate standard deviation.

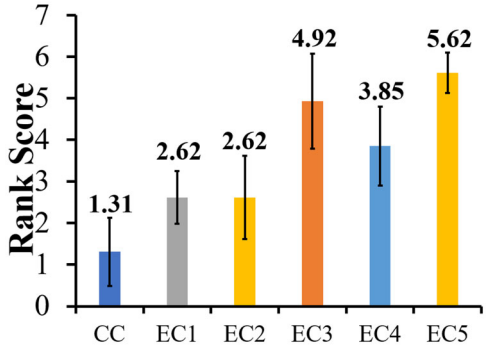

Figure 11. Rank Score for each condition.

5.3.2. Perception

We have also investigated the task load and rank score of our method using the questionnaires.

5.3.2.1. Workload. Figure 10 shows the results of the task load. The sphericity assumption is violated: $p < 0.001(S1, S2, S3)$. After applying the Greenhouse-Geisser correction, the overall ANOVA reveals significant differences between the five conditions: $(F_{2.027,38.508} = 25.364, p < 0.001)$ for S1, $(F_{2.463,46.798} = 45.621, p < 0.001)$ for S2, and $(F_{2.793,53.068} = 26.250, p < 0.001)$ for S3. Post-hoc analysis reveals that the task load of EC_1, EC_3, EC_4 and EC_5 were significantly smaller than that of *CC* for all scenes, and the task load of *EC*2 was not significantly smaller than that of *CC* for all scenes.

5.3.2.2. Rank score. Figure 11 shows the results of the rank of all conditions. The sphericity assumption is violated: $p < 0.001$. After applying the Greenhouse-Geisser correction, the overall ANOVA reveals significant differences between the five conditions: $(F_{2.516,62.902} = 72.215, p < 0.001)$. Post-hoc analysis reveals that the rank score of EC_{1-5} were significantly higher than that of *CC* for all scenes. The ranking results from top to bottom are EC_5, EC_3, EC_4, EC_1, EC_2, and *CC*.

5.4. Discussion

The results in Table 1 support ***H1:*** Participants completed the task significantly less time with EC_1, EC_3, EC_4, and EC_5 than with *CC*. So the results in Table 1 support ***H1***. There are two possible reasons: (1) If participants did not rotate and scale rotate and scale the object in time when the object is translated, the translation accuracy would be affected. Likewise, if participants did not translate in time when the object was rotated and scaled, it would also affect the rotation accuracy. Since different manipulation viewpoints are suitable for different manipulation types, *MGF* can guide the user to the appropriate manipulation viewpoint in time, allowing two users to translate, rotate and scale objects in time, thus improving efficiency. Although EC_2 can also guide the user to the appropriate manipulation viewpoint, frequent viewing of the map makes it impossible for the user to manipulate at the appropriate manipulation viewpoint in time, which affects the manipulation efficiency. (2) In *MGF* construction, some impossible manipulation viewpoints are removed, and users can avoid these impossible manipulation viewpoints for manipulation, which greatly improves the efficiency of collaborative manipulation.

EC_3 and EC_5 are faster than EC_1 and EC_4. This may be because the participants using EC_1 and EC_4 cannot accurately use teleportation to transmit to the corresponding position. Moreover, the small balls will block the user's sight and affect the manipulation.

Tables 3 and 4 do not support ***H2:*** The results in Tables 2–4 show that the null hypothesis cannot be rejected ***(H2):*** Compared to the *CC*, EC_1, EC_3, EC_4, and EC_5 significantly reduced the position error and rotation error. However, EC_2 did not significantly reduce the position error and rotation error. Compared to the *CC*, all experimental conditions did not significantly reduce the scale error. Therefore, the results in Tables 2–4 show that the null hypothesis cannot be rejected ***(H2).*** The main reason may be: *MGF* calculates the center distance and long axis angle between the target OBB and the object OBB for each sampled viewpoint in *MGF* in real-time. Furthermore, *MGF* calculates the ratio of the projected area of the target and the object. Under the guidance of those visualization methods, two users can reach the viewpoint suitable for translation and rotation manipulation at the current moment to manipulate objects in time and accurately. The EC_2 method does not significantly improve the translation and rotation accuracy because it is difficult for most people to guide to the appropriate viewpoint with the map-only guidance method, which discourages the user's enthusiasm to reach the appropriate manipulation viewpoint. The possible reason scale error does not improve significantly is that when the user RS manipulates a certain scale error, most of the effort is spent on reducing the rotation error.

The results in Figure 10 support ***H3:*** Compared with the control condition, the task load of EC_1, EC_3, EC_4, and EC_5 are reduced significantly. The main reason may be that when two participants are manipulating, the *CC* requires multiple teleportations to find a suitable manipulation viewpoint. In addition to paying attention to manipulation, the user needs to analyze which manipulation viewpoint is suitable for manipulation multiple times, resulting in two participants spending more time and fatigue. For EC_2, the participants need to analyze the positional relationship of each manipulation viewpoint and the participants' position and orientation on the map, which increases the burden and makes the participants feel impatient. Participants focus most of their efforts on manipulation accuracy for EC_1, EC_3, EC_4, and EC_5.

The results in Figure 9 show that the teleportation number of EC_{1-5} are not significantly smaller than for *CC*. Furthermore, it shows that *MGF* for collaboration manipulation is not related to the number of teleportation but is related to the manipulation viewpoint in the virtual scene. In all methods, the mean teleportation times of user RS is higher than that of user T. The possible reason is that one user is responsible for two manipulation types. As shown in Figure 11, the preferred method for most participants was EC_5. Most participants do not like the guidance method of the small balls (EC_1). The possible reason is that the participant cannot accurately reach the manipulation viewpoint with teleportation. The possible reason for only maps (EC_2) is that participants spend a long time observing the small map but had difficulty quickly establishing the correspondence between the map and the VE., and most participants do not like this method.

6. Limitations, and future work

A limitation of our method is that the computation of *MGF* does not take into account the geometric and appearance details of the manipulated object and target. During the manipulation, the more details of the object and the target are exposed, the easier it is for the user to match the object with the target. Future work is to integrate the appearance and geometric details of manipulated objects and targets into the computation of *MGF*. The larger the proportion of visible pixels with rich surface texture details and geometric features, the corresponding *MGF* value should be higher. The second limitation is that our method now does not work for cases with unknown targets since we use the target information to sample the manipulation viewpoints and compute *MGF*. So another future work is removing the requirement of the known target and exploring the gaze-based method to guide the manipulation. *MGF* calculated by our method consists of three parts: *T*, *R*, and *S*. We currently use "T" to guide users responsible for translating and use "RS" to guide users responsible for rotating and scaling. Of course, separating *R* and *S* to guide two users is also possible. If the scene is large, the user needs to walk a long distance to be guided, and the same type of users can also be multiple people. Our user study recruited 36 participants, aged 20–31, including 6 women, we mainly considered the following situations: (1) Our user study needs to be done three times, about one hour each time, and the time is longer. The participants are recruited from the students of our university. Since we are a science and engineering school, there are few girls, so there are few girls tested. Age Distribution 20–31; (2) Due to the COVID-19 epidemic, the school campus is still closed so far, and it is difficult to recruit participants of all ages in society; (3) We believe that the impact of user gender and age on manipulation is a complex topic that requires extensive user studies to model the relationships between different factors. Therefore, we plan to investigate this topic in our future work further. And by providing appropriate compensation, we can attract more participants, including more women, to participate in the experiment and enhance the diversity and inclusivity of the participants. And in future work, we will use experiments to prove the influence of different combinations of manipulation types on two-person collaboration. We did not consider the analysis of the scene as one of the factors. In the scene, whether the object being manipulated as a consideration factor affects the accuracy and efficiency of manipulation is a question worth studying. And in future work, We will study whether the two factors of the scene and manipulated object affect the accuracy and efficiency of manipulation.

7. Conclusions

We have proposed a collaborative object manipulation method guided by the manipulation guidance field to improve accuracy and efficiency. With the visualization of *MGF*, users of the different manipulation types, such as translation, rotation, and scaling, can find the locations with

better views to manipulate the objects easily. Compared with the method without *MGF*, our method has significantly reduced task completion time, position error, rotation error, and task load. We also find the hybrid visualization of color squares and the mini-map is the user's favorite *MGF* visualization. All in all, the proposed collaborative object manipulation method guided by the *MGF* has the potential to improve the usability and performance of VR applications that involve object manipulation. The concept of *MGF* and its construction method, as well as the strategies proposed to accelerate the MGF update process, can be further studied and refined to enhance the collaborative object manipulation experience in VR.

Disclosure statement

No potential conflict of interest was reported by the author(s).

Funding

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

References

Bachmann, E. R., Hodgson, E., Hoffbauer, C., & Messinger, J. (2019). Multi-user redirected walking and resetting using artificial potential fields. *IEEE Transactions on Visualization and Computer Graphics*, *25*(5), 2022–2031. https://doi.org/10.1109/TVCG.2019.2898764

Baron, N. (2016). Collaborativeconstraint: Ui for collaborative 3d manipulation operations. In 2016 IEEE Symposium on 3D User Interfaces (*3DUI*), (pp. 273–274). IEEE. https://doi.org/10.1109/3DUI.2016.7460076

Bergström, J., Dalsgaard, T.-S., Alexander, J., Hornbæk, K. (2021). How to Evaluate Object Selection and Manipulation in vr? guidelines from 20 Years of Studies. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* (pp. 1––20). IEEE. https://doi.org/10.1145/3411764.3445193

Bowman, D. A., & Hodges, L. F. (1997). An evaluation of techniques for grabbing and manipulating remote objects in immersive virtual environments. In Proceeding of the Symposium on Interactive 3D Graphics, 1997. IEEE. https://doi.org/10.1145/253284.253301

Bozgeyikli, E., Raij, A., Katkoori, S., Dubey, R. (2016). Point and teleport locomotion technique for virtual reality. In *Proceedings of the 2016 Annual Symposium on Computer-Human Interaction in Play* (p. 205–216). New York, NY, USA: Association for Computing Machinery. https://doi.org/10.1145/2967934.2968105

Chenechal, M. L., Lacoche, J., Royan, J., Duval, T., Gouranton, V., & Arnaldi, B. (2016). When the giant meets the ant an asymmetric approach for collaborative and concurrent object manipulation in a multi-scale environment. In 2016 IEEE Third vr International Workshop on Collaborative Virtual Environments (3*DCVE*), (pp. 18–22). IEEE. https://doi.org/10.1109/3DCVE.2016.7563562

Cohen, J. (2013). *Statistical power analysis for the behavioral sciences*. Academic press.

Dong, T., Chen, X., Song, Y., Ying, W., & Fan, J. (2020). Dynamic artificial potential fields for multi-user redirected walking. In 2020 *IEEE* Conference on Virtual Reality and 3d User Interfaces (vr), (pp. 146–154). IEEE.

Duval, T., Lecuyer, A., & Thomas, S. (2006). Skewer: A 3d interaction technique for 2-user collaborative manipulation of objects in virtual environments. In *IEEE Symposium on* 3d User Interfaces (3DUI'06), (pp. 69–72). IEEE. https://doi.org/10.1109/VR.2006.119

Frees, S., Kessler, G. (2005). Precise and rapid interaction through scaled manipulation in immersive virtual environments. In *IEEE Proceedings. vr 2005. Virtual Reality, 2005*. (p. 99–106). IEEE. https://doi.org/10.1109/VR.2005.1492759

Frees, S., Kessler, G. D., & Kay, E. (2007. Prism interaction for enhancing control in immersive virtual environments. *ACM Transactions on Computer-Human Interaction*, *14*(1), 2–es. https://doi.org/10.1145/1229855.1229857

Freitag, S., Weyers, B., & Kuhlen, T. W. (2016). Automatic speed adjustment for travel through immersive virtual environments based on viewpoint quality. In *IEEE Symposium on* 3D User Interfaces. IEEE.

Freitag, S., Weyers, B., & Kuhlen, T. W. (2018). Interactive exploration assistance for immersive virtual environments based on object visibility and viewpoint quality. In IEEE Virtual Reality 2018. IEEE.

Gloumeau, P. C., Stuerzlinger, W., & Han, J. H. (2020). Pinnpivot: Object manipulation using pins in immersive virtual environments. In IEEE Transactions on Visualization and Computer Graphics. Vol. *99*, (pp. 1–1). IEEE.

Grandi, J. G., Debarba, H. G., & Maciel, A. (2019). Characterizing asymmetric collaborative interactions in virtual and augmented realities. In *IEEE* Conference on Virtual Reality and 3*D* User Interfaces. IEEE.

Hart, S. (2006). Nasa-task load index (nasa-tlx); 20 years later. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, *50*(9), 904–908. https://doi.org/10.1177/154193120605000909

Hart, S., & Staveland, L. (1988). Development of nasa-tlx (task load index): Results of empirical and theoretical research. *Advances in Psychology*, *52*, 139–183. https://doi.org/10.1016/S0166-4115(08)62386-9

IBM (n.d.). *Spss software*. https://www.ibm.com/analytics/ spss-statistics-software.

Kai, R., Holtkamper, T., Wesche, G., & Frohlich, B. (2006). The bent pick ray: An extended pointing technique for multi-user interaction. In IEEE Symposium on3d User Interfaces, 2006. *3DUI* 2006. IEEE.

Kamada, T., & Kawai, S. (1988). A simple method for computing general position in displaying three-dimensional objects. *Computer Vision Graphics and Image Processing*. *41*(1), 43–56. https://doi.org/10.1016/0734-189X(88)90116-8

Khatib, O. (1985). Real-time obstacle avoidance for manipulators and mobile robots. In *Proceedings. 1985 IEEE International Conference on Robotics and Automation Vol. 2*, (p. 500–505). IEEE. https://doi.org/10.1109/ROBOT.1985.1087247

Khatib, O. (1987). A unified approach for motion and force control of robot manipulators: The operational space formulation. *IEEE Journal on Robotics and Automation*, *3*(1), 43–53. https://doi.org/10.1109/JRA.1987.1087068

Lages, W. (2016). Ray, camera, action! a technique for collaborative 3d manipulation. In *IEEE Symposium on* 3D User Interfaces. IEEE.

Liu, X., Wang, L., Luan, S., Shi, X., & Liu, X. (2022). Distant object manipulation with adaptive gains in virtual reality. In 2022 IEEE International Symposium on Mixed and Augmented Reality (*ISMAR*), (pp. 739–747). IEEE. https://doi.org/10.1109/ISMAR55827.2022.00092

Mauchly, J. W. (1940). Significance test for sphericity of a normal *n*-variate distribution. *The Annals of Mathematical Statistics*, *11*(2), 204–209. https://doi.org/10.1214/aoms/1177731915

Mendes, D., Caputo, F. M., Giachetti, A., Ferreira, A., & Jorge, J. (2019). A survey on 3d virtual object manipulation: From the desktop to immersive virtual environments. *Computer Graphics Forum*, *38*(1), 21–45.

Messinger, J., Hodgson, E., & Bachmann, E. R. (2019). Effects of tracking area shape and size on artificial potential field redirected walking. In 2019 *IEEE* Conference on Virtual Reality and 3d User Interfaces (vr), (pp. 72–80). IEEE. https://doi.org/10.1109/VR.2019.8797818

Nguyen, T.-T H., Duval, T., & Pontonnier, C. (2014). A new direct manipulation technique for immersive 3D virtual environments. International Conference on Artificial Reality and Telexistence, IEEE.

Patil, S., van den Berg, J., Curtis, S., Lin, M. C., & Manocha, D. (2011). Directing crowd simulations using navigation fields. *IEEE Transactions on Visualization and Computer Graphics*, *17*(2), 244–254. https://doi.org/10.1109/TVCG.2010.33

Pinho, M. S., Bowman, D. A., & Freitas, C. (2008). Cooperative object manipulation in collaborative virtual environments. *Journal of the Brazilian Computer Society*, *14*(2), 53–67. https://doi.org/10.1007/BF03192559

Plemenos, D., & Benayada, M. (1996. Intelligent display in scene modelling. new techniques to automatically compute good views. In *Graphicon'96*. Saint Petersburg.

Ruddle, R. (2005). *3d user interfaces: Theory and practice*. MIT Press One.

Ruddle, R. A., Savage, J. C. D., & Jones, D. M. (2002). Symmetric and asymetric action integration during cooperative object manipulation in virtual environments. *ACM Transactions on Computer-Human Interaction*, *9*(4), 285–308. https://doi.org/10.1145/586081.586084

Shapiro, S. S., & Wilk, M. B. (1965). An analysis of variance test for normality (complete samples). *Biometrika*, *52*(3–4), 591–611. https://doi.org/10.1093/biomet/52.3-4.591

Soares, L. P., Kopper, R., & Pinho, M. S. (2018). Ego-exo: A cooperative manipulation technique with automatic viewpoint control. 2018 20th Symposium on Virtual and Augmented Reality (Svr), (pp. 82–88). IEEE.

Sokolov, D., & Plemenos, D. (2005). Viewpoint quality and scene understanding. In M. Mudge, N. Ryan, & R. Scopigno (Eds.), *The 6th international symposium on virtual reality, archaeology and cultural heritage vast*. The Eurographics Association. https://doi.org/10.2312/VAST/VAST05/067-073

Sokolov, D., Plemenos, D., Tamine, K. (2006). Viewpoint quality and global scene exploration strategies. In *Grapp 2006: Proceedings of the First International Conference on Computer Graphics Theory and Applications, Setúbal, Portugal, February 25–28, 2006*.

Song, P., Goh, W. B., Hutama, W., Fu, C.-W., Liu, X. (2012). A handle bar metaphor for virtual object manipulation with mid-air interaction. In *Proceedings of the Sigchi Conference on Human Factors in Computing Systems* (p. 1297–1306). New York, NY, USA: Association for Computing Machinery. https://doi.org/10.1145/2207676.2208585

Vázquez, P., Feixas, M., Sbert, M., & Heidrich, W. (2001). Viewpoint selection using viewpoint entropy. Vision Modeling and Visualization Conference. ACM.

Wang, L., Liu, X., & Li, X. (2021). Vr collaborative object manipulation based on viewpoint quality. 2021 *IEEE* International Symposium on Mixed and Augmented Reality (Ismar), (pp. 60–68). IEEE. https://doi.org/10.1109/ISMAR52148.2021.00020

Wang, R. Y., Paris, S., & Popovic, J. (2011). 6d hands: Markerless hand tracking for computer aided design. *ACM* Symposium on User Interface Software and Technology. ACM.

Wilkes, C., Bowman, D. A. (2008). *Advantages of velocity-based scaling for distant 3D manipulation*. In *Proceedings of the 2008 ACM Symposium on Virtual Reality Software and Technology* (pp. 23–29). New York, NY, USA: Association for Computing Machinery. https://doi.org/10.1145/1450579.1450585

Zagata, K., Gulij, J., Halik, u., & Medyńska-Gulij, B. (2021). Mini-map for gamers who walk and teleport in a virtual stronghold. *ISPRS International Journal of Geo-Information*, *10*(2), 96. https://doi.org/10.3390/ijgi10020096

Zeleznik, R. C., Forsberg, A. S., Strauss, P. S. (1997). Two pointer input for 3D interaction. In *Proceedings of the 1997 Symposium on Interactive 3D Graphics* (pp. 115–120). IEEE.https://doi.org/10.1145/253284.253316

About the authors

Xiaolong Liu is a PhD student in the School of Computer Science and Engineering of Beihang University, China. His current research focuses on virtual reality, augmented reality, and HCI.

Lili Wang received her PhD degree from 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, real-time rendering and HCI.

Shuai Luan is a master student in the School of Computer Science and Engineering of Beihang University, China. His current research focuses on virtual reality, augmented reality, and HCI.