2023 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)

# Manipulation Guidance Field for Collaborative Object Manipulation in VR

Xiaolong Liu[1] Shuai Luan[1] Lili Wang[1;2] * Chan-Tong Lam[3]

[1]State Key Laboratory of Virtual Reality Technology and Systems, Beihang University, Beijing, China
[2]Peng Cheng Laboratory, Shengzhen, China
[3]Macao Polytechnic University, Macao, China.

2023 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW) | 979-8-3503-4839-2/23/$31.00 ©2023 IEEE | DOI: 10.1109/VRW58643.2023.00199

## 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. We first introduce the concept of *MGF* and its construction method. Then we propose two strategies to accelerate *MGF* updating process. After that, we propose a collaborative manipulation method to manipulate objects using the guidance of *MGF*.

**Index Terms:** Virtual reality—Collaboration—Object manipulation—Guiding field;

## 1 Introduction

Object manipulation is a fundamental interaction in product design, 3D object modeling and virtual object assembly in virtual reality (VR) applications. The efficiency and accuracy of object manipulation directly impact the application performance. Object manipulation in VR includes single-user manipulation [3] and multi-user manipulation [8]. 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 [4].

In order to improve the efficiency and accuracy of collaborative manipulation, some methods assign different manipulation types to different users. In the early work, the user's location is fixed, and his manipulation type does not change after pre-specification [2, 6]. Recently, the user has been allowed to move in the manipulation process [5, 7]. 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. Based on this, we introduce the concept of the manipulation guidance field. And to guide the user more reasonably and efficiently, two problems need to be addressed. The first one is how to find viewpoints suitable for a given manipulation type in the virtual scene when the object is manipulated. The second problem is how to guide the user to the appropriate viewpoint to manipulate the object.

In this paper, we introduce a collaborative method guided by the manipulation guidance field (*MGF*) to improve manipulation accuracy and efficiency in multi-user VR applications. 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*.

*Lili Wang is corresponding author: wanglily@buaa.edu.cn

## 2 Object Manipulation Guidance Field

In this section, first, we introduce the concept of the object manipulation guidance field in Sect. 2.1, then the *MGF* construction and updating methods are given in Sect. 2.2. An optimization method is provided to accelerate the updating in Sect. 2.3.

### 2.1 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*.

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

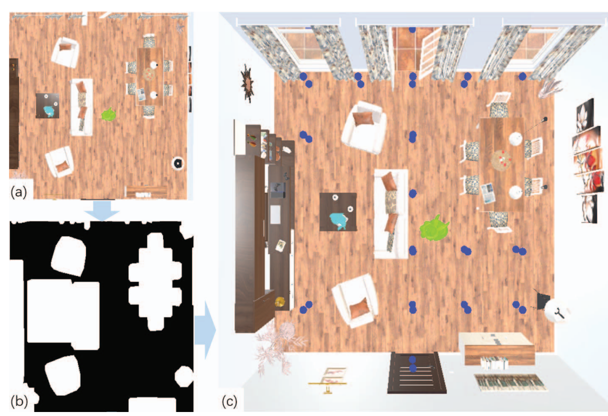


Figure 1: (a) is the top view of Scene. (b) is the walkable area *WA*. (c) is the a third view of Scene, and the blue balls are the sampled viewpoints in (c).

### 2.2 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.

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. Based on *Navimesh*, we get the walkable area *WA* of the entire scene, as shown in Fig. 1 (b). We sample the locations inside *WA* uniformly with the horizontal interval 1.5*m*. On

979-8-3503-4839-2/23/$31.00 ©2023 IEEE
DOI 10.1109/VRW58643.2023.00199

each location, two viewpoints are placed with heights $1.0m$ (corresponding to the user's crouching posture) and $1.7m$ (corresponding to the person's normal standing posture). We call these sampled viewpoints $SV$, as shown in Fig. 1 (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 Fig. 2. For each output frame, we rendered an image from this camera, and the resolution is $32 \times 32$. And we use this image to calculate the value of T/R/S.

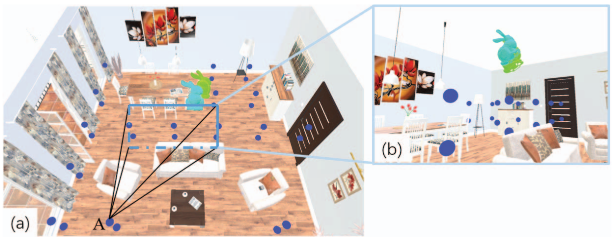


Figure 2: (a) is a third view of Scene. (b) is a view of viewpoint A in $SV$. The blue balls are the sampled viewpoints.

## 2.3 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.

**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 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. 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. In our implementation, we set $d_{Thres}$ to $3m$, $Thres_1$ to 0.01, $Thres_2$ to 0.005, $Thres_3$ to 0.002.

**Strategy 2. Updating with a time interval**

In order to reduce the time cost of ***M(T,R,S)*** update, we reduce the update frequency. This is because the user manipulation action is not so fast, 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.

# 3 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. When the user translates the object, the user focuses on the spatial location of the object and the target. When the user rotates and scales the object, the user focuses on the details of the object and the appearance of the object. So the visualization of '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 design three ways to visualize $MGF$ : color balls, color squares and mini-map.

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 5Hz. 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 teleportation [1] method to go to the yellow location.

# 4 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. However our method now does not work for the cases with unknown targets since we use the target information to sample the manipulation viewpoints and compute $MGF$. So future work is removing the requirement of the known target and exploring the gaze-based method to guide the manipulation.

## Acknowledgments

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

## References

[1] E. Bozgeyikli, A. Raij, S. Katkoori, and R. Dubey. Point and teleport locomotion technique for virtual reality. In *Proceedings of the 2016 Annual Symposium on Computer-Human Interaction in Play*, CHI PLAY '16, p. 205–216. Association for Computing Machinery, New York, NY, USA, 2016. doi: 10.1145/2967934.2968105

[2] M. L. Chenechal, J. Lacoche, J. Royan, T. Duval, V. Gouranton, and B. Arnaldi. When the giant meets the ant an asymmetric approach for collaborative and concurrent object manipulation in a multi-scale environment. pp. 18–22, 2016.

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

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

[5] W. Lages. Ray, camera, action! a technique for collaborative 3d manipulation. In *3d User Interfaces*, 2016.

[6] M. S. Pinho, D. Bowman, and C. Freitas. Cooperative object manipulation in collaborative virtual environments. *Journal of the Brazilian Computer Society*, 14(2):53–67, 2008.

[7] L. P. Soares, R. Kopper, and M. S. Pinho. Ego-exo: A cooperative manipulation technique with automatic viewpoint control. In *2018 20th Symposium on Virtual and Augmented Reality (SVR)*, 2018.

[8] 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)*, pp. 60–68, 2021. doi: 10.1109/ISMAR52148.2021.00020