Group-based Object Alignment in Virtual Reality Environments

Rongkai Shi
Xi’an Jiaotong-Liverpool University
Suzhou, China
rongkai.shi19@student.xjtlu.edu.cn

Wolfgang Stuerzlinger
Simon Fraser University
Vancouver, Canada
w.s@sfu.ca

Jialin Zhang
Xi’an Jiaotong-Liverpool University
Suzhou, China
jialin.zhang21@student.xjtlu.edu.cn

Hai-Ning Liang∗
Xi’an Jiaotong-Liverpool University
Suzhou, China
haining.liang@xjtlu.edu.cn

ABSTRACT
Group-based object alignment is an essential manipulation task, particularly for complex scenes. In conventional 2D user interfaces, such alignment tasks are generally achieved via a command/menu-based interface. Virtual reality (VR) head-mounted displays (HMDs) provide a rich immersive interaction experience, which opens more design options for group-based object alignment interaction techniques. However, object alignment techniques in immersive environments are underexplored. In this paper, we present four interaction techniques for 3 degrees-of-freedom translational alignments: AlignPanel, AlignWidget, AlignPin, and AlignGesture. We evaluated their performance, workload, and usability in a user study with 20 participants. Our results indicate different benefits and drawbacks of these techniques for group-based alignment in immersive systems. Based on the findings, we distill a set of design choices and recommendations for these techniques in various application scenarios.

CCS CONCEPTS
• Human-centered computing → Virtual reality; Empirical studies in interaction design; Interaction techniques.

KEYWORDS
virtual reality, manipulation task, group object alignment, interaction techniques, user studies

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

SUI ’22, December 1–2, 2022, Online, CA, USA. ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3565970.3567682

∗Corresponding author (haining.liang@xjtlu.edu.cn)
single object to another. To the best of our knowledge, there is little existing work on object alignment techniques in immersive VR environments, and the interaction techniques to support group-based alignments are underexplored. We aim to fill this research gap in this paper and, as a first step, we focus on 3-DOF translational alignments.

Towards this goal, we first specify the 3-DOF translational group-based alignment problem space and derive three criteria that the design of the interaction techniques for alignment tasks should meet. Based on these design criteria, we then introduce four new techniques—AlignPanel, AlignWidget, AlignPin, and AlignGestures for VR head-mounted displays (HMDs). Then, we conduct a user study with 20 participants to compare the user performance, workload, and experience with the four proposed techniques. Finally, we present a discussion of the results and point to design recommendations and directions for future work before concluding the paper.

In short, the contributions of this paper include: (1) The design of four interaction techniques for 3-DOF group-based object alignment tasks in VR HMDs and other immersive systems. (2) An empirical study with twenty participants, pointing to design choices and recommendations for the practical use of the proposed techniques.

2 RELATED WORK

2.1 Object Manipulation in VR

Virtual Hand is one of the primary input paradigms in immersive systems, e.g., with VR HMDs [18, 23]. Common input devices such as hand-held controllers and optical tracking modules afford 6-DOF tracking, comprising 3 translational axes and 3 rotational axes, and provide direct and natural interactions using virtual hand in a VE similar to the interactions in the real world [30]. Researchers have developed enhanced or extended versions of the virtual hand technique to make it more functional in different contexts (e.g., Go-Go [27], PRISM [8], and HOMER [40]). Nowadays, the virtual hand approach or its extensions are commonly used for object manipulation tasks in immersive VR environments.

Lavioila et al. [18] defined selection, positioning, rotation, and scaling as the four canonical manipulation tasks for 3D UIs. Within VR contexts, translation and rotation are the two most frequent manipulation tasks investigated in the literature [2]. There have been lots of discussions on whether to integrate or separate the DOF controls for object translation and rotation tasks. On the one side, researchers proposed to control the translational and rotational DOF at the same time because integrated manipulation control has a parallel, interdependent structure and retains the “naturalness” and “intuitiveness” of people’s behaviors [12, 23, 27, 38]. On the other side, results from empirical studies have shown that 6-DOF manipulation, though suitable for fast coarse transformation, cannot offer precise manipulation [24]. Separating the translational and rotational controls can improve accuracy—for example, completing the task in a single dimension (i.e., 1-DOF) individually can ensure higher levels of precision [24, 25, 37]. Given the benefits of separating DOF, researchers have considered it as a feature when designing mid-air interaction techniques (e.g., DS3 [22], MAIOR [25], and Plane, Ray, and Point [14]).

In this paper, we explore mid-air interactions for aligning virtual objects in an immersive VR system. An object alignment task, by its nature, typically has an implicit demand for precision. Thus, we considered DOF separation and, as a first step to fill the research gap, we focus on 3-DOF translational alignments. In addition, when multiple alignment requirements exist, we decompose the 3-DOF task into several 1-DOF sub-tasks to help users achieve the highest possible precision.

2.2 2D and 3D Alignment Tools

Alignment or other positioning manipulation techniques have been well studied in 2D UIs. The alignment approaches in these tools can be broadly categorized into snapping-based, command-based, and constraint-based. Command/menu-based techniques are most common in current commercial applications [29]. In the 1990s, Briar [11] combined the snap-dragging technique [4] and constraints to support 2D drawings, where the snapping function would assist the user in making quick alignments according to specified constraints. The Beyond Snapping approach [5] helped users make persistent and tweakable alignments with user-defined StickLines in graphics editing applications. GACA [43], on the other hand, used commands to achieve group-aware arrangements of graphical elements. In multi-touch displays, Frisch et al. [9] implemented Grids & Guides to support continuous and flexible alignments by snapping the objects to interactive grids and multi-touch alignment guides. More recently, they proposed NEAT to support alignment and other layout tasks using bi-manual gestures [10]. Rock & Rails [39] is another gesture-based manipulation system. It facilitates precise graphical layouts with a set of hand-shape gestures in combination with direct manipulation of objects.

Unlike the broad exploration of such interaction techniques in 2D environments, little prior work has been conducted on the design and development of alignment tools for 3D environments, especially for immersive environments. Song et al. [33] designed a handle bar metaphor to support object manipulation for desktop VR using freehand gestures. With a handle bar, users can manipulate and align multiple objects along a straight line (i.e., a bar). The Context-based 3D grids [1] enable users to perform precise mid-air manipulations in AR HMDs by displaying a reference grid relative to the hand. Hayatpur et al. [14] proposed Plane, Ray, and Point, which utilized symbolic gestures to specify (temporary) constraints for precise spatial manipulations. It enables quick alignment of virtual objects against plane constraints.

2.3 Multiple Object Manipulation

Many interaction techniques have been proposed for manipulating a single object at a time. While repeating the same manipulation for multiple objects is possible, it would also be more time-consuming and tedious than grouping the objects first and manipulating them together, especially when there are many objects to be aligned [21]. Dual constraints [35] afforded quick creation of object groups and direct manipulation of the groups in desktop VE. By dual constraining the objects nearby, the objects in the group can then be moved, rotated, and regrouped together [32, 35]. When working with virtual objects that have some association with one another, the hierarchical relationship of the group implicitly defined the
constraints, and thus users can interact with the object group or its sub-group(s) (e.g., [42]). In the **Beyond Snapping** approach, users can snap the target graphical objects on pre-defined **StickyLines** and then apply alignment or distribution relationships to them [5]. Similarly, **handle bar** attaches virtual objects and manipulate them together [33]. In these two metaphors, the grouping and the group-based manipulations are achieved via explicitly formed line constraint(s). With **Plane, Ray, and Point**, users can create a plane constraint to align the objects in immersive VR environments [14]. When using these techniques, users need to first specify the constraint(s) explicitly. The constraint(s) is then used for forming a group and performing follow-up manipulations. In our work, we focus on a different problem space, where users are aiming to align multiple objects based on their implicitly formed constraints.

### 3 PROBLEM FORMULATION

In this paper, we focus on 3-DOF translational group-based alignments in VR environments. For simplicity, we investigate spatial alignments along the 3D cardinal directions, as shown in Fig. 2. Rotating the coordinate system then naturally affords alignment to other axes, which is a general strategy used in commercial applications (e.g., [16]). In our work, we define the term ‘group-based alignment’ as adjusting the positions of the selected objects precisely according to the furthest surface in a specific direction (align left, right, top, bottom, front, or back) or their geometric centers (align middle and along x-, y-, or z-axis). The alignment is discrete and sequential, which means that each alignment is an individual 1-DOF operation, and the target position in an alignment is always calculated based on the current position of the object group.

In addition, when applying an alignment constraint to the target objects, the objects can be **intersecting** or **non-intersecting**, as shown in Fig. 3. An intersecting spatial relationship ignores objects’ physical information and, as such, the objects can partially intersect with or contain others. In contrast, a non-intersecting relationship disallows the intersections between the objects, which means at best two objects can touch each other (but not intersect). Both spatial relationships are considered in this work. For simplicity, we define that an intersection in a direction can only be established if the objects have been aligned in that direction. In other words, an intersection can only happen after alignment.

A general procedure for performing a group-based alignment involves two main phases: (1) **Selection phase**: In this phase, the user first needs to select a group of target objects. (2) **Alignment phase**: the user needs to apply one or more align constraint(s) iteratively to the selected objects until all the requirements are satisfied. In this work, our primary focus is on the second alignment phase.

### 4 DESIGN CRITERIA

Given the challenge of 3-DOF group-based alignment in VR as defined above, we specified three design criteria (denoted as C#) that such an alignment technique must meet to be effective.

- **C1: Applicability.** The proposed interaction techniques should be applicable within current VR HMDs and immersive systems and should suit commercial VR modeling or content creation applications (e.g., Blocks [16] or Sketchbox [17]). Thus, we considered and designed the techniques that are usable with the ubiquitous hand-held controllers, which are represented as virtual hands in the VE.

- **C2: Near-field Interaction.** The proposed technique for 3-DOF alignments should always be within arm’s reach so that a user can easily activate it. A near-field interaction technique can reduce human errors compared to interacting at a distance [19, 40]. Furthermore, it avoids unnecessary body movements, leading to improved efficiency and reduced physical demands.

- **C3: Disambiguation.** Efficiency, learnability, and memorability are three usability criteria that any interactive system should meet [28] (pp. 25). A technique with clear operational affordances supports efficient interaction and helps users learn how to use it quickly. We thus made the interaction mechanism consistent in each technique, while allowing the operations towards different alignments to vary naturally. For example, though the gestures in a gesture-based technique may (spatially) vary for different alignments, their semantic interpretations share the same concept.

### 5 DESIGN OF TECHNIQUES

In this work, we designed and implemented four techniques for group-based 3-DOF alignment in virtual environments: (1) **AlignPanel**, (2) **AlignWidget**, (3) **AlignPin**, and (4) **AlignGestures**. All assume that the user has first selected the group of objects that are to
be aligned. Fig. 4 shows an example of using these techniques to complete an align-right task.

5.1 Align Panel

AlignPanel is a 2D menu-based interface, with icons and texts and uses a layout typical for desktop applications. We created icons with 3D illustrations to represent the various alignments along the x, y, and z-axis, as shown in Fig. 4a. A user can then click on a button to achieve the corresponding alignment. Once selected, a button is highlighted to also indicate the current alignment state in the user interface.

5.2 Align Widget

AlignWidget is a widget-based technique. Overall, its appearance is similar to a typical 3D widget with virtual handles [6, 24]. It consists of six outer handles, three inner axis sticks, and a central cube. For each of the three axes, the two outer handles and the axis stick represent the two ‘side’ alignments and the central alignment, respectively. Tapping on a handle or a stick then aligns the selected objects based on their spatial arrangement relative to the widget. For example, tapping on the right-most handle will align the objects to the right, as shown in Fig. 4b and e. The handles and sticks are colored by axes: red for the x-axis, green for the y-axis, and blue for the z-axis. An activated alignment is indicated by highlighting the borders of the touched handle or stick. To prevent users from false activations of central alignments, we used sticks through the widget center instead of handles, as this approach increases the distances between each widget component. Additionally, we also placed an inactive grey cube in the center. Although AlignWidget is movable, our current implementation for the user study cannot be rotated. This ensures that the representations of the alignment will always match the global axes, thereby minimizing confusion.

5.3 Align Pin

Our AlignPin technique is an adapted version of a technique from TinkerCAD [15], a 3D modeling web application. When a user opens AlignPin after selecting a group of objects, AlignPin creates a bounding box for the group and shows three interactive pins along three edges of that bounding box. The user can then touch one of the corresponding pin heads to align the objects. We use 3D pins instead of the 2D ‘flat’ pins that TinkerCAD uses because the 3D ‘style’ makes the pins not only more visible from different viewing directions but also easier to interact with in the VE. The indication for an active alignment is the same as in AlignWidget. To meet our design criterion C2, we created a hand-held replica of the AlignPin widget. While this looks on the surface similar to a World-In-Miniature approach [34], we only replicate the widget and not the objects to avoid occlusions (see Fig. 4c). Similar to AlignWidget, AlignPin allows translational movements but not rotational movements.

5.4 Align Gesture

With AlignGesture, the user performs a hand gesture to align the selected objects. We identified nine hand gestures that match the alignment requirements, as shown in Fig. 5. We aimed to make the gestures semantically meaningful and easy to understand. The underlying logic is to move the hand toward the implicitly constructed constraint surface. To perform a side alignment, a user then only needs to keep one hand reasonably still (which then represents the constraint) and move the other hand close to it [26]. On the other hand, the constraint for middle alignments lies between the two hands. Thus, moving two hands toward each other along the target axis specifies an align-middle action. In our implementation and to prevent false positives, the user needs to hold the grip buttons on both controllers simultaneously to perform an AlignGesture. In addition, the user needs to release the buttons after the gesture, to be able to perform potential other alignment actions (as described in Section 3).

We fine-tuned our gesture recognition through a pilot study with 5 participants, which collected hand movement data. The participants performed each gesture 5 times. Based on the collected data, we were able to identify good thresholds for hand movement distances to recognize the gestures reliably.

6 USER STUDY

The aim of our study was to compare and evaluate the four proposed techniques for 3-DOF group-based alignment in VR environments. We followed the VR object selection and manipulation study checklist provided by Bergström et al. [2] to design and report the study.

6.1 Participants

We recruited 20 participants (9 females, 11 males) with ages between 19 to 31 years (mean = 22.60, SD = 2.50). All were university students and volunteered to participate in this user study. Based on the results from a questionnaire applied before the experiment, all of them reported that they were right-handed and had normal or correct-to-normal vision. Fifteen participants were familiar or very familiar with VR HMDs, and ten participants were familiar or very familiar with 3D content design or creation applications (such as CAD, 3ds Max, Photoshop, etc.). Fourteen participants used a VR HMD every month or week. Five participants used 3D content design or creation applications on a monthly basis, and seven used them on a weekly or daily basis.

6.2 Apparatus and Experiment Setup

An Oculus Quest 2 standalone VR HMD was used to display the virtual environment (VE). This HMD has a display resolution of 1832 × 1920 per eye, 89° horizontal field of view, and a 120Hz refresh rate. The HMD was connected to a Windows 10-based desktop with an Intel i7-8700K CPU @ 3.70GHz, an NVIDIA GeForce GTX 1080 Ti GPU, and 16 GB of RAM. Two Oculus controllers were used for interaction with the VE. Our software was developed using Unity (version 2021.3.36f1c1) with the SteamVR Plugin (version 2.7.3).

The study was conducted in an empty room. The participants were standing while performing the tasks.

Note that the gestures for the alignments along the x-axis are typically unique due to the nature of people’s hands (i.e., left and right hands). In contrast, the gestures for alignment in the y- or z-axis can be performed in two different ways. For example, a user can position their right hand below and move the left hand close to it, or vice versa, to perform an align-bottom action.
Figure 4: For an align-right task, this figure shows the user interfaces of the four investigated techniques: (a) AlignPanel, the user clicks on the button; (b) AlignWidget, the user taps on the handle; (c) AlignPin, the user taps on the pin head; and (d) AlignGesture, the user performs the hand gesture. The resulting alignment is shown in the bottom panel (e). Note that we are showing a non-intersecting alignment, thus the medium-grey front-most bottom block aligns only to the dark grey block to avoid interpenetration.

Figure 5: The gesture set for AlignGesture, where each of the nine gestures represents one of the potential 3-DOF object alignments in a way that matches the geometric relationships of the alignment operation.

In the experiment, participants saw five objects that they were asked to align and a task specification to specify the alignment requirements (see Fig. 6a and b). To simplify the task, we used cuboids that were within view and within reach from the initial position. To eliminate a potential experimental confound, participants could not directly select a cuboid to translate, rotate, or scale. We used text descriptions in the task specification rather than images or 3D models to reduce the cognitive load for planning and decision-making because such cognitive load could lead to reduced user performance and also negatively affect the experience of using a technique, which could bias the results.

Participants interacted with the VE via the handheld controllers. To undo an action, users needed to press Button ‘A’ on the right controller, which would move all objects to their previous positions, i.e., to their state before the last action. For AlignPanel, AlignWidget, and AlignPin, Button ‘Y’ on the left controller toggled the corresponding widget on or off, while Button ‘X’ on the left controller toggled the widget following the participant’s left, non-dominant hand movements, which we identified in our pilot study as a usable method that worked well. By default, the widgets for these three techniques were off and, when turned on, would move with the participant’s left hand. As mentioned previously, for AlignGesture a user needed to press and hold the grip buttons of two controllers to perform the corresponding alignment action.

6.3 Design, Task, and Procedure

Our study used a 4 × 2 within-subjects design with two independent variables: Technique (AlignPanel, AlignWidget, AlignPin, and AlignGesture), and Alignments (one-time and three-time). Alignments varies the number of alignments that participants needed to perform in a task: one-time means one requirement (e.g., middle alignment along the z-axis in Fig. 6a), while three-time would require three different alignments, one for each axis (e.g., middle alignment along the z-axis, then bottom alignment along the y-axis, and finally middle alignment along the x-axis; see Fig. 6b). The alignment tasks were presented randomly.

The task required participants to align the objects according to the task specification shown above the scene. As mentioned in Section 3, all the cuboids were already selected at the beginning of a trial. A task then ended when the participants had completed the required alignment action(s). The tasks involved both non-intersecting and intersecting alignments. For an intersecting alignment task, the only alignment in the one-time task and the third alignment in the three-time task involved an intersecting alignment (Fig. 6b).

The order of Technique conditions was counterbalanced using a Latin-Square design. Following general experimental design recommendations to present tasks in order of complexity, the one-time tasks were presented before the three-time tasks for each technique. Each Alignments condition involved 6 alignment tasks, including 3 intersecting alignment tasks and 3 non-intersecting alignment tasks. In total, we collected data from 20 participants × 4 Technique × 2 Alignments × 6 alignment tasks = 960 tasks.

Participants were first informed of the purpose of the study and completed a pre-questionnaire collecting their demographic and
past experience information. Then, they were briefed about the VE and the four new interaction techniques, followed by a 10-minute ‘warm-up’ session with the VR device to get accommodated to the VE and the control schemes for each technique. The experiment involved four sessions, corresponding to the four techniques. For each session and to ensure that participants understood the task and current interaction technique well, four training trials, including one time intersecting and two time intersecting alignment trials, were given before the formal trials. After each session, we asked participants to record their subjective feelings about the just-used technique through ratings that covered different aspects (introduced in the next section). In addition, the participants were encouraged to take a short break between two sessions. At the end of the experiment, the participants received a short interview.

The whole experiment lasted approximately 35 minutes for each participant.

6.4 Evaluation Metrics

We measured the performance of the techniques in terms of number of failures, and completion time. We defined trials where participants needed to resort to undo actions as failed trials. Number of failures is the number of failed trials for each Technique×Alignment condition. Completion time is the time to complete a given alignment task in seconds. We did not consider the failed trials in this measurement because the number of undo actions can confound this time and the reasons for participants performing undos were not always clear. We recorded the time from when the participant first pressed the grip button(s) to perform an AlignGesture or pressed the ‘Y’ button for the first time to show the widget for the other three techniques until the required alignment had been achieved.

In addition, we collected participants’ subjective ratings after each session. We used the Raw NASA-TLX [13] to assess the required workload to complete a given task and a positive System Usability Scale (SUS) [20] to evaluate the usability of the techniques. At the end of the experiment, we conducted a semi-structured interview where we asked participants about their experience with the new techniques with respect to different aspects, including (1) the ranking of the techniques based on their overall preference and the reasons for giving this ranking, if they could express them; (2) the strengths and weaknesses of each technique; and (3) any possibilities that they could think of for using or extending the techniques for 6-DOF alignments (i.e., manipulations involving 3-DOF rotations).

6.5 Hypotheses

Based on our literature review, our design process, and pre-experiment pilot trials, we formulated the following hypotheses:

- **H1.** Participants’ performance in terms of the number of failures and completion time in the given tasks would not vary significantly among the four proposed techniques. In the design of the techniques, we followed our three design criteria. We hypothesize that all four techniques are usable and support efficient interaction (C1, C2). For each technique, the actions for different alignments are easy to distinguish (C3).

- **H2.** Participants experience more workload to complete the given tasks with AlignGesture than with the other three techniques. Participants interact with AlignPanel, AlignWidget, and AlignPin through a virtual tool using simple actions (i.e., tapping a button). Compared to them, AlignGesture might experience relatively higher mental effort since participants have to recall and perform the corresponding gesture.

- **H3.** The perceived usability of AlignPanel is higher than the other three techniques. Given that AlignPanel is adapted from a typical 2D UI, participants might be more familiar with it and thus find it easy to learn and intuitive to use, both of which are important measures of the technique’s usability [20]. Thus, AlignPanel could be rated more usable for our well-controlled but reasonably straightforward tasks.

7 RESULTS

The results were analyzed and visualized using R (version 4.2.1) and RStudio (version 2022.02.3).

7.1 Objective Measures

Among the 960 trials we recorded, we considered 104 trials as failed (completed with undo), and these were counted as the number of failures. For the remaining 856 successful trials, we removed 6 trials (0.70%) as outliers due to extreme completion times. For the number of failures and completion time, we performed two-way repeated measures (RM-)ANOVA tests. According to the QQ plots, completion time was normally distributed, while the number of failures was not. Thus, we applied Aligned Rank Transform [7, 41] to the number of failures before the RM-ANOVA tests. If a significant difference was found, pairwise comparisons were conducted with Bonferroni corrections. When the assumption of sphericity was violated, we report the degrees of freedom with Greenhouse-Geisser corrections ($\epsilon < 0.75$ in our cases). We report performance separately for each type of Alignment task, as the interaction between Alignment and Technique may not provide meaningful insights.

7.1.1 Number of failures. The average number of failures was statistically different for Technique ($F_{3,133} = 4.515, p = 0.005, \eta_p^2 = 0.092$) and Alignment ($F_{2,133} = 19.624, p < 0.001, \eta_p^2 = 0.129$). There was no significant interaction between Technique and Alignment in the number of failures ($F_{3,133} = 2.130, p = 0.099, \eta_p^2 = 0.046$). Post-hoc tests did not identify significant differences between Technique in one-time tasks or three-time tasks ($p > 0.05$). Fig. 7a shows the results.

7.1.2 Completion time. There was a significant main effect of Alignment on completion time ($F_{1,19} = 291.098, p < 0.001, \eta_p^2 = 0.96$).
The sub-scales of the NASA-TLX were scaled to a range from 0 to 100, the same as their weighted overall scores. We applied Aligned Rank Transform [7, 41] for non-parametric factorial analyses for the NASA-TLX scores and SUS scores.

### 7.2 Subjective Measures

7.2.1 NASA-TLX. There was a significant effect of Technique on overall workload scores ($F_{3,57} = 6.778, p < 0.001, \eta^2_p = 0.162$). For the sub-scales of the NASA-TLX, we found significant effects for mental demand ($F_{3,57} = 7.969, p < 0.001, \eta^2_p = 0.167$), physical demand ($F_{3,57} = 6.627, p < 0.001, \eta^2_p = 0.137$), temporal demand ($F_{3,57} = 4.799, p = 0.005, \eta^2_p = 0.150$), and effort ($F_{3,57} = 5.283, p = 0.003, \eta^2_p = 0.155$). Fig. 8a shows the results and any significant differences from pairwise comparisons.

7.2.2 SUS. There was a significant effect of Technique on overall SUS scores ($F_{3,57} = 7.488, p < 0.001, \eta^2_p = 0.176$). Pairwise comparisons revealed that the perceived usability of AlignPanel was significantly higher than AlignWidget ($p = 0.025$) and AlignGesture ($p = 0.008$). Fig. 8b shows the results.

7.2.3 Rankings. Fig. 9 summarizes the subjective rankings in terms of overall preference. We present and discuss all other interview responses in the Discussion section.
we discuss the techniques separately in this section. Global axes might make the farthest parts of the widgets easier to interact with, they also shared the concern that such rotation may generate confusion. In terms of the positive aspects of this technique, two participants (P1, P3) mentioned AlignPanel as a good presentation of the 3D relationships.

8.1.3 AlignPin. Interestingly, AlignPin was slow in one-time tasks but fast in three-time tasks (see Fig. 7b). One possible reason is that participants needed to first locate the correct pin head among the separated axes, which is relatively time-consuming compared to the other techniques in an one-time task; while since it was “easy to understand” (N = 4) with “representative visuals” (N = 5), participants were able to complete multiple alignments without a complicated thought process, which yielded a shorter completion time in the three-time tasks. On the whole, AlignPin was highly usable, required a low workload, and was preferred by the participants for doing a group-based alignment task.

8.1.4 AlignGesture. In contrast to AlignPin, AlignGesture was fast in the one-time tasks but slow in the three-time tasks. Unlike the previous three techniques, AlignGesture did not require participants to locate and select a target alignment option, so it could be performed quickly when users accurately recalled the correct gesture. Though the overall workload of AlignGesture was only significantly lower than AlignPanel (which did not support H2), we found that it was harder to learn and to remember how to use it, compared to the other techniques. This led to high mental demand and effort, as well as low usability (see Fig. 8). Especially when doing an iterative three-time task, the cognitive load was relatively high for recalling the gesture(s). Nevertheless, we found seven participants ranked it as the most preferred technique and three ranked it as the second-most preferred, which together is half the size of our participant pool. Participants gave this high ranking mainly because they believed if they used it for a longer time, AlignGesture would be the fastest and the most natural technique (N = 7).

8.2 Design Choices and Recommendations

Current VR modeling applications, such as Sketchbox [17] and Blocks [16], use an in-hand menu-based interface, which makes them easy to integrate AlignPanel. However, when asked about its use for 6-DOF alignment tasks, twelve participants said it was not suitable or could not be used. Indeed, using a 2D menu to represent and control complex 3D relationships in immersive environments might be less intuitive. Thus, AlignPanel is only appropriate when the use case requires 3-DOF translational alignments and when the users’ view matches or roughly matches the global axes.

For the 3D widget-based techniques, all participants agreed that both AlignWidget and AlignPin could be used for 6-DOF alignment tasks as long as the rotation of the widget matches the rotation of the object group. In addition, it is also possible to use these widgets directly to control the rotation of the object group, simply by rotating the widget. Based on our results, we suggest AlignPin to be used in practical applications, while AlignWidget is not recommended due to its relatively low usability. Compared to AlignPanel, AlignPin could have broader uses in immersive VR environments.

Based on the collected feedback, AlignGesture is a ‘controversial’ technique. There is no denying that AlignGesture has a high learning
cost, even if a subset of users may feel that such gestural input is natural and intuitive. As for using it for 6-DOF alignment tasks, participants felt it was easier to use for such scenarios because they could perform the gesture along the rotated axes. However, three participants (P4, P10, P16) raised concerns about the precision of hand movement recognition. Given the above aspects, we do not recommend using AlignGesture or only integrating it as an additional option for long-term users who prefer to use it.

Though most participants chose not to toggle the current alignment widget following their hand movement, we still recommend giving this choice to users in real application scenarios. Our experimental trials involved two types of alignment tasks, with and without intersections. We did not observe a difference between these types during the experiment or from the collected results, which was in line with our expectations as repeated actions were necessary for these conditions. Thus, it is possible to use the same mechanism in real scenarios. Another alternative is to make the type of intersection a system option so that users can choose to activate or deactivate the intersection behavior for the following sets of manipulations.

This work explored techniques for 3-DOF group-based object alignment where the alignment constraints are implicitly formed based on the object group. By adding a step for specifying alignment constraint(s) before the alignment phase (see Section 3), we believe that our proposed techniques are still applicable to other types of alignments as long as all the selected objects need to match the same alignment goal.

Though our user study was conducted using a VR HMD, we expect that our results and the derived design choices and recommendations could inspire and be beneficial to the design and development of applications in other immersive systems, such as CAVE or AR HMDs. Our results could also generalize to other types of applications in immersive systems which involve similar group manipulations. For example, VR world-building games can benefit from our research, as our techniques enable players to precisely align large numbers of structural elements, which supports the quick creation and modification of larger structures. Another example could be collaborative productivity applications, where users might want to align cluttered notes, to-do items, mind map blocks, or other visual elements from different collaborators, to make the hierarchy or flow better visible. We thus believe that our techniques allow efficient and accurate group-based alignments of visual elements across many VR applications.

8.3 Limitations and Future Work
We identified the following limitations, which could point to directions for future work. First, we used a fairly simple strategy for gesture recognition, which also yielded reasonably robust recognition. Although we did not observe any recognition errors during the user study, other approaches such as Dynamic Time Warping [3] could further improve the precision and enable its use in a larger variety of contexts. Second, as the first exploration of group-based object alignment in VR, we used a controlled experimental design and limited the task to the alignment phase with translational movements. Future work could evaluate the techniques by giving participants more freedom to play with the techniques and also to test performance with more complex alignment tasks, including 6-DOF ones, to see if the techniques provide a smooth workflow for such scenarios too. Furthermore, it is worth investigating whether or how our techniques could coexist with other VR selection and manipulation techniques that are optimized for performance and complex scenarios [44–46], or allow for rapid mode-switching interaction [31, 36]. Third, we would like to enhance and embed our techniques into existing VR modeling applications [16, 17] to compare their performance with a baseline approach in real use cases, which at the moment typically involves the manipulation of only a single object at a time. Finally, most of our participants were familiar with VR headsets. A more varied participant pool could provide additional insights, especially when the tasks and/or the techniques become more challenging, as mentioned above.

9 CONCLUSION
In this work, we explored group-based alignment in immersive virtual reality (VR) systems, primarily focusing on three degrees of freedom (3-DOF) translations. We first identified three design criteria that a technique must meet to address the alignment of a group of objects in VR, and then proposed four interaction techniques: AlignPanel, AlignWidget, AlignPin, and AlignGesture. They were evaluated via a user study with 20 participants in terms of their performance, workload, and usability. Our results showed that using AlignPanel and AlignPin induced less workload and were considered highly usable. AlignWidget received negative feedback and thus is not suggested for practical use. Though participants generally rated AlignGesture worse in workload and usability, they also saw some benefits in this technique. Based on these findings, we provided several design recommendations for using the four alignment techniques in VR applications. Our alignment techniques and the recommendations derived from the user study can help researchers and designers develop future VR object alignment techniques that can be integrated into a variety of application scenarios.

ACKNOWLEDGMENTS
The authors want to thank the participants for their time and the reviewers for the insightful comments and suggestions that helped improve our paper. This work was supported in part by Xi’an Jiaotong-Liverpool University (XJTLU) Key Special Fund (#KSF-A-03) and XJTLU Research Development Fund (#RDF-17-01-54).

REFERENCES