© 2026 IEEE. This is the author's version of the article accepted for publication in IEEE TVCG.
The final version is available at: https://doi.org/10.1109/TVCG.2026.3680697

Exploring Freehand Selection Techniques of Polyhedron Faces in VR Environments

Yifan Qi *
Institute of Software
Chinese Academy of Sciences
Beijing, China

Xuning Hu *
Institute of Software
Chinese Academy of Sciences
Beijing, China
The Hong Kong University of Science and Technology (Guangzhou)
Guangzhou, China

Xinan Yan
Institute of Software
Chinese Academy of Sciences
Beijing, China
The Hong Kong University of Science and Technology (Guangzhou)
Guangzhou, China

Wenxuan Xu
Dartmouth College
Hanover, USA

Hao Zhang
Institute of Software
Chinese Academy of Sciences
Beijing, China

Hai-Ning Liang
The Hong Kong University of Science and Technology (Guangzhou)
Guangzhou, China

Jin Huang ‡
Institute of Software
Chinese Academy of Sciences
Beijing, China

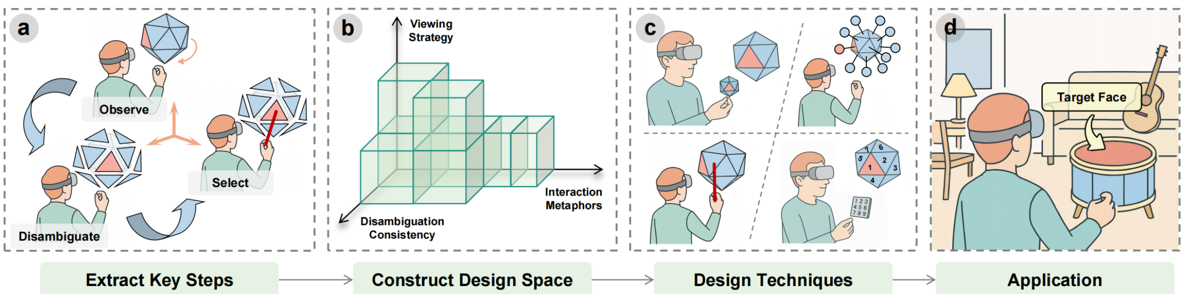

Fig. 1. Overall workflow with four stages for designing and evaluating polyhedral face selection techniques: (a) identifying the key steps involved in polyhedral face selection(see Sec. 3), (b) constructing a three-dimensional design space based on key steps(see Sec. 3), (c) designing interaction techniques for selected regions of the design space and evaluating their performance across different scenarios(see Sec. 4), and (d) building complex application cases to assess the practical applicability of the techniques(see Sec. 7).

Abstract— Virtual reality (VR) allows users to observe and manipulate 3D geometry from multiple viewpoints. Most VR selection work, however, optimizes techniques for selecting entire objects. Selecting a single face on a polyhedron remains underexplored and is more challenging because the interaction must act on a local component while preserving the object's global structure. We introduce a design space tailored to this task with three dimensions: viewing strategy, disambiguation consistency, and interaction metaphor. Guided by this space, we design eight freehand techniques for polyhedral face selection. A within-subjects study with 16 participants evaluates these techniques across polyhedral complexity (two radii; face counts 4, 6, and 12). The results identify three complementary top techniques, reveal tradeoffs between viewing choices and geometric preservation, and yield concrete guidelines for matching techniques to target geometry and task demands. A follow-up study with complex, realistic models confirms the robustness and practical usability of the three techniques. Together, these contributions shift attention from whole object selection to precise component selection in VR and provide actionable methods for 3D modeling, assembly, and texturing.

Index Terms—Virtual Reality, Selection, Modeling, Polyhedron, Precise Interaction.

1 Introduction

Selection is a core operation in Virtual Reality (VR) systems, and recent advances in tracking and input have enabled increasingly precise and intuitive interaction. Yet most selection techniques still optimize for choosing *entire* objects in complex scenes. Many application scenarios, however, require interaction with constituent parts, such as specific faces, of a single object, including object assembly [11], 3D modeling [72], and painting and texturing [69]. Contemporary VR selection research predominantly addresses small targets [59, 60], occluded targets [18, 68], or densely arranged targets [16, 4], typically treating objects as atomic units. Although components can be treated as independent targets in interaction tasks, faces are not spatially independent: they share boundaries with adjacent faces and are tightly coupled to the object's global geometry and orientation. Techniques in this vein, including bubble cursor snapping and Bayesian pointer inference, are inherently biased toward whole-object selection and do not directly support *selection at the face level* within a single object [24]. Consequently, fine-grained selection within a target's internal structure remains underexplored.

Beyond incompatibilities at the technique level, polyhedral selection also differs in interaction logic. Polyhedra exemplify this gap: they are integrated 3D structures composed of polygonal faces joined by shared edges. In polyhedral selection tasks, users often rotate, scale, or otherwise manipulate the entire object to reveal specific faces while maintaining a holistic understanding of its geometry, rather than focusing on isolated or occluded items as in conventional cluttered-scene selection. Region-enclosure techniques (e.g., lasso) and scene-

*These authors contributed equally to this work.
†Corresponding author (e-mail: huangjin@iscas.ac.cn).
‡This work was supported in part by the National Natural Science Foundation of China (Grant Nos. 62332017 and 62402486), the Key Project of the Institute of Software, Chinese Academy of Sciences (ISCAS-ZD-202401), and the Key Project of the Key Laboratory of System Software, Chinese Academy of Sciences (ZD202502).

separation approaches (e.g., smash) can improve precision for single targets, but they conflict with the need to preserve structural coherence and to minimize disruption to object integrity during face-level operations [64]. These characteristics motivate a reexamination of whether existing techniques extend to this setting and the development of methods tailored to the cohesion, adjacency, and orientation-dependent visibility inherent to integrated 3D objects.

In desktop modeling software such as Maya and Blender, sophisticated methods for selection at the face level (e.g., brush selection, regional cursors) enhance efficiency and user experience [3, 5]. VR's immersive presentation strengthens spatial perception, and six degrees of freedom (6DoF) input affords fine, direct manipulation. Despite VR modeling tools such as Gravity Sketch[1] and Google Blocks[2] (released in 2016 and 2017, respectively), face selection in VR remains rudimentary: it relies mainly on standard controller interactions without optimization for precise selection at the face level. Controller use during fine operations can occlude targets and consume the field of view. Moreover, interacting with complex polyhedral geometry via ray casting or virtual hands often yields ambiguous or erroneous selections. These limitations indicate the need for dedicated freehand techniques for face selection explicitly tailored to VR environments.

To address face selection in VR, we identify three primary challenges: (1) *Visibility.* Target faces are often invisible from the current viewpoint, requiring frequent viewpoint changes or object rotations that reduce interaction efficiency. (2) *Structure preservation.* Decomposing or rearranging the object (e.g., Expand [12] or GridWall [68]) can disrupt users' mental models of polyhedral structures. (3) *Precision limits of hand input.* Hand interactions in VR, constrained by the Heisenberg effect [7], generally cannot match the accuracy of mouse input, leading to more selection errors and a higher workload. Regarding interaction modalities, advances in freehand VR have underscored the naturalness and immersive potential of hand interactions [10]. However, mainstream VR modeling applications (e.g., Gravity Sketch, Masterpiece X[3]) still rely primarily on controller input, leaving efficient freehand face selection comparatively underexplored.

Given this gap, we propose a structured framework explicitly designed for efficient polyhedral face selection using freehand interaction. The framework comprises three stages: (1) *viewpoint adjustment*, (2) *disambiguation*, and (3) *precise face selection*. Building on this framework, we define a three-dimensional design space spanned by viewing strategies, disambiguation consistency, and interaction metaphors (see Sec. 3). Within this space, and informed by prior techniques and the properties of polyhedra, we design and develop eight optimized freehand techniques with bimanual gesture coordination to facilitate face selection (see Sec. 4). We evaluate these techniques across varying levels of polyhedral complexity, measuring selection time, error rate, and subjective responses, including NASA-TLX, UEQ-S, and semi-structured interviews (see Sec. 5). We then refine the top-performing techniques based on empirical findings and conduct a second study to validate their practicality and effectiveness in realistic scenarios (see Sec. 7).

The main contributions of this work are: (1) a three-dimensional design space tailored for precise face selection on complex polyhedra and related integrated 3D structures; (2) the design and systematic evaluation of eight freehand interaction techniques for face-level selection; and (3) validation of their practical applicability through real-world cases, yielding insights to guide future research and tool design for VR-based polyhedral face selection.

2 Related Work

This section first reviews prior work on freehand interaction for target selection and explains the advantages of hand-based input in VR. It then synthesizes existing techniques designed for whole-object selection across varied scenarios. Finally, it analyzes their potential and limitations for enabling face-level selection on polyhedral objects.

2.1 Freehand-based Interaction

Freehand-based interaction is controller-free and has emerged as a prevalent paradigm in VR [44], offering a high level of immersion [10, 50]. This input modality maps physical hand movements directly onto virtual hands [71], enabling naturalistic interaction afforded by six-degrees-of-freedom (6-DoF) hand positions. As such, it is suitable for interacting with 3D objects such as polyhedra in VR [52]. In addition, despite criticisms of inefficiency and inaccuracy [9], freehand-based interactions significantly enhance immersion, presence, and effective communication [10]. Consequently, this modality has been extensively utilized in various scenarios like user interface navigation [67], target selection [13], text entry [48], and gaming [35, 51].

Limitations arise in selecting distant targets with freehand interactions, prompting various solutions. To enhance reachability, the GO-GO technique has been proposed [49, 21], where the virtual arm extends non-linearly as the user's physical hand moves outward to reach distant objects. Gesture-based object selection is another common approach, with Wu et al. proposing a set of gestures for common interaction tasks [62]. Additionally, virtual pointing techniques overcome reachability constraints [19], where users emit rays from virtual hands to select intersected objects [63]. Despite these positive approaches, issues remain, such as the "Heisenberg effect", where unintended hand movements occur during input [7, 8], inaccurate selections in cluttered scenes [47], and mismatches between ray-casting and visual focus [1]. To retain the immersiveness of freehand input while improving efficiency, we treat supportive techniques as a key design dimension for stabilizing or augmenting hand-based selection. In this study, we compare virtual-hand, ray-casting, and support-enhanced techniques under varying levels of polyhedral complexity, aiming to clarify their respective strengths and limitations for freehand-based object selection.

2.2 Target Selection

Numerous studies have optimized target selection tasks, primarily by enhancing selection accuracy [70, 39, 30, 56], and there is also work that models user accuracy as a function of task difficulty [28, 29]. Relatedly, Huang et al. modeled uncertainty in moving-target selection across 1D/2D pointing and crossing tasks [32, 33, 31]. With ray-casting, research has explored altering ray origins, such as emitting rays from eyes [2], or dual-ray techniques [46]. Additionally, researchers have explored modifying the shape of the selection ray to improve accuracy. Liang et al. [40] suggested using cones, Forsberg et al. [20] introduced adjustable apertures via hand rotation. Moreover, some studies have explored viewpoint optimization, such as scaling objects based on distance to selection rays [2, 60] or presenting additional viewpoints [25]. With virtual-hand selection, Scott et al. [21, 22] proposed PRISM, dynamically adjusting the C/D ratio to improve accuracy and speed.

Previous research has also addressed diverse scenarios. For small, distant objects, techniques like expanding target width were proposed by Kabbash and Buxton [36], Wei et al. [59] introduced a Fan Mechanism, and Weller et al. [60] dynamically scaled potential targets. For dynamic objects, Krüger et al. [38] evaluated user intentions based on object scores, and Boffi et al. [6] automatically filtered irrelevant objects through dynamic scoring. In dense environments, Delamare et al. [16] proposed Multi-Finger-Bubble, and Bhowmic et al. [4] introduced Tiny Hands for selecting small targets. Occlusion scenarios also received significant attention. For example, Elmqvist and Tsigas [18] reviewed numerous occlusion management methods, and Yu et al. [68] compared various occlusion-selection techniques to find some optimal ones.

Nevertheless, these techniques mainly focus on selecting entire objects, with limited exploration into the precise local selection of the constituent elements of these objects. Jackson et al. [34] systematically compared three manual techniques for convex region selection on surface geometries, representing an exploration of intra-object local

[1]https://gravitysketch.com/ [Accessed: 2025-03-04]
[2]https://github.com/googlevr/blocks [Accessed: 2025-03-04]
[3]https://www.masterpiecex.com/ [Accessed: 2025-11-04]

© 2026 IEEE. This is the author's version of the article accepted for publication in IEEE TVCG.
The final version is available at: https://doi.org/10.1109/TVCG.2026.3680697

selection. However, that study addresses region-level selection on continuous surfaces, focusing on contour-based interactions, whereas our work targets face-level selection on discrete, well-defined polyhedral structures. This fundamental difference leads to distinct interaction and judgment requirements.

Overall, prior work on target selection in VR can be broadly grouped into: (1) techniques optimizing whole-object selection accuracy under challenging conditions such as distance, density, motion, and occlusion; (2) methods that adapt control–display mappings or selection volumes to improve robustness; and (3) limited investigations into intra-object selection, primarily focusing on continuous surface regions rather than discrete components.

2.3 Modeling in 3D Immersive Environment

VR enhances spatial perception, which is particularly beneficial for interacting with three-dimensional structures and models by enabling intuitive and natural interactions. In architectural design [55], it facilitates the development of interactive visualization systems based on Building Information Modeling, allowing users to freely navigate and explore 3D architectural models at a true-to-scale ratio [15]. In education, VR enables instructors to interact with polyhedra of various shapes, providing more accessible teaching methods for spatial cognition-related subjects such as geometry [37]. In 3D modeling, VR offers a more intuitive modeling approach, allowing users to sketch directly in three-dimensional space to generate models [42, 41].

Despite the extensive applications of 3D modeling, interaction methods, especially techniques that allow for precise selection, are underexplored. For instance, Wang et al. [57] studied interactions with exhibits in virtual museums but focused only on entire-object selection. This highlights a gap in research regarding the precise selection of specific parts of 3D models and similar objects, a critical requirement in tasks such as box modeling [72], where precise selection of polyhedral faces is essential. This paper addresses this research gap by exploring interaction techniques specifically designed for precise face selection, focusing on polyhedral objects in VR environments.

3 Design Space

Prior work characterizes target selection as a five-step workflow: *observation*, *disambiguation*, *activation*, *selection*, and *feedback* [68, 65]. Our study focuses on *observation*, *disambiguation*, and *selection*—the stages most consequential for spatial interaction in VR, where activation and feedback often arise implicitly from direct manipulation and system constraints. We therefore operationalize a three-dimensional design space along the following axes: *Viewing Strategy*, *Disambiguation Consistency*, and *Interaction Metaphors*. These axes are directly derived from the observation, disambiguation, and selection stages of the workflow described by Yu et al. [68], with each axis isolating a key design decision that shapes how these stages are realized in freehand face-level selection.

3.1 Viewing Strategy

Observation determines how users acquire an informative pose relative to the target face; viewing techniques strongly influence spatial perception [43], interaction experience, and cognitive load [61]. We consider three established camera-control metaphors [58], but operationalize *Viewing Strategy* with two levels in our study.

Eyeball-in-hand. The viewpoint (virtual camera) is aligned with hand motion; users adjust position and orientation through hand rotation. In Unity, holding `Alt` and dragging the mouse follows this metaphor, reorienting the camera without altering scene elements[4]. This affords precise, target-centered inspection of a single face.

Scene-in-hand. The environment (or object) is treated as a manipulable artifact; users rotate the scene rather than the camera. In Unity/Blender, engaging an object-rotation modifier (e.g., `E`) and dragging rotates an object about its axes[5], supporting object-centered exploration and inspection of enclosed details.

Flying-vehicle control (excluded). Aircraft-like navigation uses nonlinear velocity mapping [14] and, in Unity, is exemplified by holding the right mouse button with `WASD`[6]. While efficient for traversing large spaces, its coupling of translation and rotation makes it difficult to keep a single target centered while performing smooth, object-relative rotations [58]. As our task requires face-centered observation rather than scene traversal, we do not include this level.

In summary, *Viewing Strategy* in our design space has **two levels**: *eyeball-in-hand* and *scene-in-hand*.

3.2 Disambiguation Consistency

Ambiguity in polyhedral face selection arises from two sources. First, occlusion and grazing angles hinder visibility of the target face and are primarily mitigated by the viewing strategy. Second, tight face adjacency increases confusability during fine-grained selection, which is the focus of disambiguation. Unlike choosing among multiple objects, selecting components within a single object imposes intra-object constraints.

Prior work proposes disambiguation in dense scenes [16, 54], but many techniques do not transfer to component-level selection. Progressive pipelines assume separable objects, while unfolding/restructuring methods (e.g., Expand [12], GridWall [68]) violate the requirement to preserve the polyhedron's geometric integrity during selection.

We therefore define *disambiguation consistency* as the extent to which object attributes are preserved during intra-object disambiguation. Two levels are considered: **Maintain** (preserve geometry/topology/pose; ambiguity reduced via appearance cues or proxies) and **Disrupt** (temporarily relax attributes—e.g., explode or scale—to reduce occlusion/adjacency). Geometry-preserving approaches stabilize users' mental maps but may leave some ambiguity unresolved; attribute-relaxing approaches can reduce ambiguity but introduce remapping overhead and must be carefully bounded.

3.3 Interaction Metaphors

In VR, object selection is primarily realized through *virtual pointing* (ray casting) and *virtual hands* (near-field touch/manipulation). Ray casting extends reach but is sensitive to hand jitter for small or distant faces due to the Heisenberg effect [7, 8], causing misselections driven by system interpretation rather than user intent. Virtual hands offer natural, precise control within reach [45, 46], but require bringing the target—or a proxy—into the near field.

To improve stability and reduce intent–interpretation mismatch, we also consider *assistive tools* that convert far-field, continuous aiming into near-field or discrete actions (e.g., on-object widgets, numeric keyboards, guided frames) [17]. These tools can mitigate jitter sensitivity and increase precision at the cost of additional interaction steps and learning, and their applicability may be task-specific.

4 Potential Techniques

Grounded in the design space defined above, we instantiate a baseline and eight optimized techniques for polyhedral face selection. Existing VR modeling tools (e.g., Adobe Medium[7] and Google Blocks) commonly rely on controller-based ray casting. We adapt this paradigm to freehand interaction as our baseline, then vary the three axes—*Viewing Strategy*, *Disambiguation Consistency*, and *Interaction Metaphors*—to derive the remaining techniques in Fig. 2. As freehand input represents an emerging and increasingly adopted interaction modality in VR, emphasizing greater directness, embodiment, and natural hand motion compared to controller-based input

[4] `https://docs.unity3d.com/6000.0/Documentation/Manual/PositioningGameObjects.html` [Accessed: 2025-11-04]

[5] `https://docs.unity3d.com/6000.0/Documentation/Manual/SceneViewNavigation.html` [Accessed: 2025-11-04]

[6] `https://docs.unity3d.com/6000.0/Documentation/Manual/SceneViewNavigation.html` [Accessed: 2025-11-04]

[7] `https://www.adobe.com/products/medium.html` [Accessed: 2025-11-04]

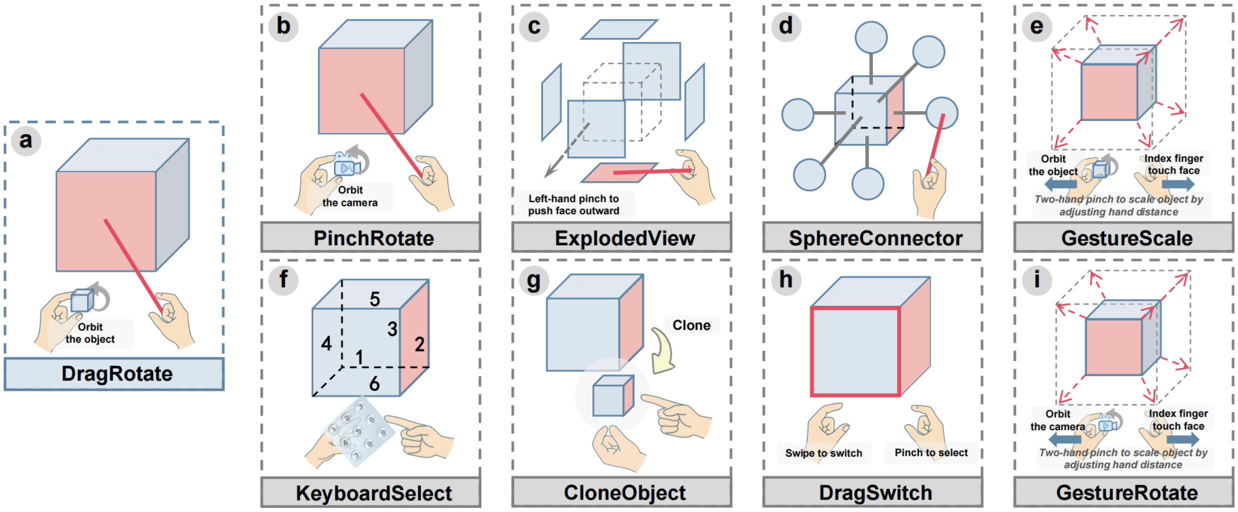

Fig. 2. Baseline DragRotate (a) and design-space–instantiated techniques: PinchRotate (b), ExplodedView (c), SphereConnector (d), GestureScale (e), KeyboardSelect (f), CloneObject (g), DragSwitch (h), and GestureRotate (i).

(see Sec. 4), all techniques in this work were designed and evaluated within the same input modality to ensure a fair and controlled comparison.

DragRotate (DR): *(Viewing: scene-in-hand; Disambiguation: maintain; Interaction: ray casting)*. Users rotate the polyhedron with the left hand and cast a selection ray with the right hand, confirming with a pinch gesture (Fig. 2a). DR introduces no assistive elements or additional visuals, reflecting mainstream VR workflows and serving as our baseline.

PinchRotate (PR): *(eyeball-in-hand; maintain; ray casting)*. Users manipulate the viewpoint via left-hand pinch to rotate the camera while simultaneously aiming a right-hand ray to select the target face (Fig. 2b). The technique preserves disambiguation consistency during selection.

SphereConnector (SC): *(scene-in-hand; maintain; ray casting)*. Users rotate the polyhedron with left-hand pinch and aim with a right-hand ray (Fig. 2d). To enlarge selectable areas and mitigate jitter-induced errors (Heisenberg effect) under tight face adjacency, a small proxy sphere is placed at each face centroid, offset by distance δ along the face normal $\hat{\mathbf{n}}$, with radius proportional to face area ($k \cdot \sqrt{A_{face}}$). Users may select via the proxy spheres or directly on faces; geometry is preserved.

ExplodedView (EV): *(scene-in-hand; disrupt; ray casting)*. EV reduces occlusion by temporarily separating faces while rotating the object (Fig. 2c). With each left-hand pinch, face i translates along its normal and slightly rotates about the Y-axis:

$$\mathbf{p}_i(k) = \mathbf{R}_Y(\theta) \cdot (\mathbf{p}_i(k-1) - \mathbf{p}_{center}) + \mathbf{p}_{center} + \delta \cdot \mathbf{n}_i \tag{1}$$

where $\mathbf{p}_i(k-1)$ is the prior position, δ the translation step, $\mathbf{n}_i$ the normal, and $\mathbf{R}_Y(\theta)$ the rotation matrix. Spatial ordering is preserved, but geometric integrity is temporarily disrupted to enhance visibility; selection uses ray casting.

CloneObject (CO): *(scene-in-hand; maintain; virtual hands)*. Inspired by dense-selection techniques [68], a miniature duplicate appears at the user's fingertip through a left-hand pinch (Fig. 2g). The duplicate keeps a constant scale and rotates with the palm, with orientation:

$$\mathbf{R} = [\mathbf{y} \times \mathbf{z} \quad \mathbf{y} \quad \mathbf{z}] \tag{2}$$

and basis vectors:

$$\mathbf{z} = \frac{\mathbf{p}_i - \mathbf{p}_p}{|\mathbf{p}_i - \mathbf{p}_p|}, \quad \mathbf{y} = \frac{(\mathbf{p}_p - \mathbf{p}_w) \times (\mathbf{p}_i - \mathbf{p}_w)}{|(\mathbf{p}_p - \mathbf{p}_w) \times (\mathbf{p}_i - \mathbf{p}_w)|} \tag{3}$$

where $\mathbf{p}_w$, $\mathbf{p}_i$, and $\mathbf{p}_p$ are wrist, index-root, and pinky-root coordinates. Users directly touch the duplicate with the right-hand fingertip to select the corresponding face on the original model; geometry is maintained.

GestureScale (GS): *(scene-in-hand; disrupt; virtual hands)*. Users perform a two-hand pinch to scale while rotating freely (Fig. 2e). The scale updates with hand-distance variation:

$$\mathbf{S}_{scaled} = \mathbf{S}_{initial} \cdot \left(\frac{|\mathbf{p}_L - \mathbf{p}_R|}{|\mathbf{p}_{L_0} - \mathbf{p}_{R_0}|} \cdot k_{scale} \right) \tag{4}$$

with $\mathbf{S}_{scaled}$ and $\mathbf{S}_{initial}$ as current and initial sizes, and $\mathbf{p}_{L_0}, \mathbf{p}_{R_0}, \mathbf{p}_L, \mathbf{p}_R$ the initial/current hand positions. Users then select by direct touch. Disambiguation consistency is intentionally disrupted to reduce adjacency and occlusion.

GestureRotate (GR): *(eyeball-in-hand; disrupt; virtual hands)*. Similar to GS in using virtual-hand scaling, but GR rotates the *viewpoint* (not the object) via an eyeball-in-hand strategy (Fig. 2i). Disambiguation consistency is not maintained.

KeyboardSelect (KS): *(scene-in-hand; maintain; assistive tools)*. Each face displays a unique numeric identifier. Users rotate the polyhedron with the left hand, then bring the left palm toward themselves to activate a virtual keyboard (Fig. 2f). The right hand confirms selection by pressing the corresponding number. Geometry is preserved while far-field aiming is converted into a discrete, stable action.

DragSwitch (DS): *(scene-in-hand; maintain; assistive tools)*. A movable red frame provides visual guidance across adjacent faces (Fig. 2h). Left-hand pinch gestures shift the frame incrementally using

$$\text{bestMatchFace} = \arg\max_i \left(\mathbf{h} \cdot \frac{\mathbf{C}_i - \mathbf{C}_c}{|\mathbf{C}_i - \mathbf{C}_c|} \right) \tag{5}$$

where $\mathbf{h}$ is the normalized hand-motion direction, and $\mathbf{C}_c, \mathbf{C}_i$ are current and adjacent face centers. A right-hand pinch confirms selection. Geometry remains intact while guidance reduces ambiguity.

The distribution of all techniques across the three design-space dimensions is summarized in Tab. 1. A step-by-step demonstration for each technique is provided in the supplementary video.

5 User Study 1

We explored and evaluated nine techniques for polyhedral face selection in VR, aiming to identify effective design features that are useful across scenarios.

5.1 Participants and Apparatus

We recruited 16 participants from local universities (10 females and 6 males), aged between 19 and 27 years (mean = 22.1, SD = 2.2),

© 2026 IEEE. This is the author's version of the article accepted for publication in IEEE TVCG.
The final version is available at: https://doi.org/10.1109/TVCG.2026.3680697

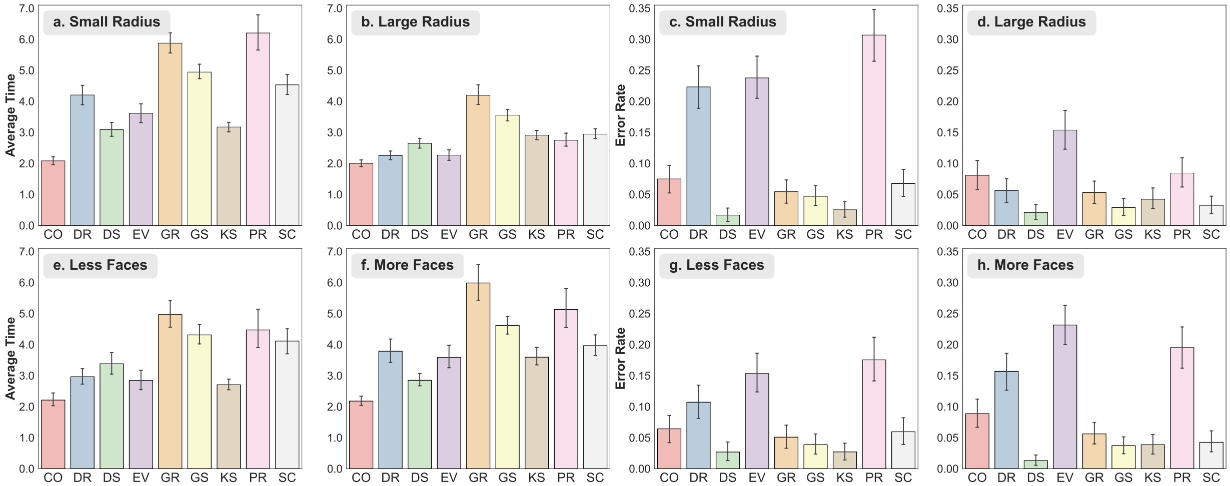

Fig. 3. Objective measurement plots: (a–d) show the average selection time and error rate under different radius conditions. (e–h) present the average selection time and error rate for polyhedra with fewer and greater numbers of faces. Error bars indicate the standard error. Statistical significance is reported in Sec. 5.5 and detailed in Sec. C

.

Techniques	Viewing		Disambiguation		Interaction		
	Eyeball in Hand	Scene in Hand	Maintain	Disrupt	Ray casting	Virtual Hands	Assistive Tool
DragRotate		●	●		●		
PinchRotate	●		●		●		
SphereConnector		●	●		●		
CloneObject		●	●			●	
ExplodedView		●		●	●		
GestureScale		●		●		●	
GestureRotate	●			●		●	
DragSwitch		●	●				●
KeyboardSelect		●	●				●

Table 1. Patterns of the techniques across the three dimensions of the design space.

with diverse academic backgrounds (e.g., computer science, microelectronics, journalism). All participants had normal or corrected-to-normal vision, with no reports of color vision deficiencies, and all identified as right-handed. Nine participants reported regular use of VR head-mounted displays. The experimental program was developed in C# on the Unity3D platform and executed on a high-performance PC equipped with an Intel Core i9 processor and an NVIDIA GTX 4060 GPU. The VR experiment was streamed from this PC to a Meta Quest 3 headset via Meta Quest Link, running in PC VR mode. The Quest 3 is equipped with dual LCD displays, providing a per-eye resolution of approximately 2064×2208 and supporting refresh rates of 120 Hz[8]. End-to-end system latency ranged from 14.4 ms to 220.5 ms [23]. Hand tracking was implemented using the Meta XR Interaction SDK (v64). Pinch gestures were detected via the built-in `GetFingerIsPinching(HandFinger)` API provided by the SDK[9].

5.2 Study Design and Task

We adopted a $9 \times 3 \times 2$ within-subjects design with factors TECHNIQUE (9), NUMBER OF POLYHEDRON FACES (4, 6, 12), and POLYHEDRON RADIUS (0.02 m, 0.2 m). Before the formal study, a 6-participant pilot verified technique usability and confirmed a substantial difficulty separation between the two radii (see Sec. B). To ensure a fair comparison across techniques and to mitigate learning effects, technique order was randomized per participant; within each technique, the (FACES×RADIUS) blocks were further randomized. In each block, target faces were sampled *without replacement* until all faces of that polyhedron had been selected. Across all techniques and conditions, participants selected all faces (4, 6, and 12), resulting in a total of 6,336 trials. The task was to select a highlighted target face on a regular polyhedron. At the start of each trial, a blue polyhedron appeared 1 m in front of the participant, with the target face highlighted in red. Participants used the assigned technique to select the target via freehand interaction, prioritizing accuracy. If a non-target face was selected, the polyhedron remained, and the attempt was logged as an error; upon a correct selection, the system generated a new polyhedron for the next trial. To support spatial orientation under the two viewing strategies (scene-in-hand vs. eyeball-in-hand), the environment included a floor and non-interactive props as stable reference cues.

5.3 Evaluation Metrics

We consider a set of dependent variables encompassing both objective metrics and subjective measurements.

5.3.1 Objective Measurements

For the objective measurements, we used per-participant, per-condition means for statistical analysis. The objective metrics included:

- ***Selection Time*** (s): the time taken to select the target in each trial correctly.
- ***Error Rate***: the proportion of incorrect selections in each condition (e.g., confirming selection when the ray/hand/key action does not correspond to the target face or its proxy).

5.3.2 Subjective Measures

We also evaluated the techniques based on subjective measurements, including user experience and preference rankings. The subjective measures included:

- ***User Experience Questionnaire (UEQ-S)*** [53]: the 8-item short form reporting Pragmatic Quality, Hedonic Quality, and Overall experience.
- ***NASA-TLX*** [26]: a six-dimensional measure of perceived workload.
- ***Preference Ranking and Interviews***: participants ranked the techniques (overall and by modality) and explained their preferences, perceived usability, encountered issues, and suitable application contexts.

5.4 Procedure

Participants first completed a demographic questionnaire. Then, they were introduced to the experimental task and VR device , consistent

[8] https://www.meta.com/quest/quest-3/ [Accessed: 2025-12-28]
[9] https://developers.meta.com/horizon/reference/unity/v77/class_o_v_r_hand/ [Accessed: 2025-12-28]

with prior studies, were instructed to complete the trials as quickly and accurately as possible [66]. After, they put on the VR headset and familiarized themselves with the virtual environment. Next, participants proceeded to the formal experiment within a fixed physical area. The experiment was structured into nine sessions, each corresponding to the evaluation of one technique. In each session, there were three phases: practice, formal trials, and post-trial evaluation. During practice, participants were taught how to use the technique and were allowed to practice freely until they felt comfortable with it. They then proceeded to complete a series of formal trials. Throughout the experiment, they were allowed to take breaks between conditions to minimize fatigue. At the end of the study, we collected user feedback using the NASA-TLX, UEQ-S, and ranking questionnaires. The entire experiment lasted about 40 minutes per participant.

5.5 Results

We collected a total of 6,336 trials. After excluding 114 trials (1.8%) that deviated more than three standard deviations from the mean (mean ± 3 SD), we analyzed 6,222 valid records. Normality was assessed using the Shapiro–Wilk test and Q–Q plots, confirming that selection time followed a normal distribution, whereas error rate and subjective measures (NASA-TLX and UEQ-S) violated normality assumptions. Data for error rate and subjective scores underwent pre-processing through the Aligned Rank Transform (ART) procedure. Subsequently, two-way repeated-measures ANOVAs (RM-ANOVAs) were conducted, with Greenhouse–Geisser corrections applied where necessary and Bonferroni-adjusted pairwise comparisons used for post-hoc analysis [27]. Effect sizes were quantified using partial eta squared (η_p^2), with values of 0.01, 0.06, and 0.14 conventionally interpreted as small, medium, and large effects, respectively. To support design-oriented interpretation, we analyzed technique effects separately within each face-count and radius condition, enabling direct comparison under representative interaction scenarios rather than fitting a single unified factorial model.

5.5.1 Selection Time

Fig. 3(a, b, e, f) shows mean selection time by technique across face-count and radius conditions. RM-ANOVA revealed a significant main effect of Technique on selection time for small radius ($F_{3.662,54.930} = 23.021,\ p < .001,\ \eta_p^2 = .605$) and large radius ($F_{1.820,27.297} = 15.630,\ p < .001,\ \eta_p^2 = .510$), as well as for low-face-count polyhedra ($F_{3.942,59.133} = 18.149,\ p < .001,\ \eta_p^2 = .547$) and high-face-count polyhedra ($F_{3.261,48.909} = 22.899,\ p < .001,\ \eta_p^2 = .604$).

Across all conditions, CloneObject yielded the shortest mean selection time. Beyond CloneObject, the fastest technique depended on condition. *Small radius:* DragSwitch and KeyboardSelect were faster, each significantly outperforming GestureRotate, GestureScale, PinchRotate, and SphereConnector. *Large radius:* DragRotate and ExplodedView were faster; specifically, DragRotate was significantly faster than GestureRotate ($p = .009$), GestureScale ($p < .001$), and KeyboardSelect ($p = .004$). *Low face count:* ExplodedView and KeyboardSelect were significantly faster than GestureRotate, GestureScale, PinchRotate, and SphereConnector. *High face count:* DragSwitch was faster and significantly outperformed GestureRotate ($p < .001$), GestureScale ($p < .001$), PinchRotate ($p = .001$), and SphereConnector ($p = .005$).

5.5.2 Error Rate

Fig. 3(c, d, g, h) shows error rates across all conditions. RM-ANOVA indicated a significant main effect of Technique for small radius ($F_{8,120} = 39.684,\ p < .001,\ \eta_p^2 = .726$) and large radius ($F_{8,120} = 11.097,\ p < .001,\ \eta_p^2 = .425$), as well as for low face count ($F_{8,120} = 14.799,\ p < .001,\ \eta_p^2 = .497$) and high face count ($F_{8,120} = 32.743,\ p < .001,\ \eta_p^2 = .686$).

Small radius: DragRotate, ExplodedView, and PinchRotate showed relatively higher error rates; PinchRotate was highest, not differing from ExplodedView or DragRotate, but significantly higher than all other techniques ($p < .001$). *Large radius:* ExplodedView had the highest error rate, significantly higher than DragSwitch ($p < .001$), GestureRotate ($p = .021$), GestureScale ($p = .001$), KeyboardSelect ($p < .001$), and SphereConnector ($p = .002$). *Low face count:* DragSwitch and KeyboardSelect achieved the lowest error rates, each significantly lower than ExplodedView and PinchRotate. *High face count:* DragSwitch again had the lowest error rate, significantly lower than CloneObject ($p = .002$), DragRotate ($p < .001$), ExplodedView ($p < .001$), and PinchRotate ($p < .001$).

5.5.3 Nasa-TLX

Different techniques had significant effects on *Mental* ($F_{8,120} = 11.189,\ p < .001,\ \eta_p^2 = .427$), *Physical* ($F_{8,120} = 46.620,\ p < .001,\ \eta_p^2 = .757$), *Performance* ($F_{8,120} = 15.957,\ p < .001,\ \eta_p^2 = .515$), *Effort* ($F_{8,120} = 40.948,\ p < .001,\ \eta_p^2 = .732$), and *Frustration* ($F_{8,120} = 21.580,\ p < .001,\ \eta_p^2 = .590$). Detailed scores and pairwise results are provided in Sec. D. The overall NASA-TLX score ($F_{8,120} = 45.929,\ p < .001,\ \eta_p^2 = .754$) is shown in Fig. 4.

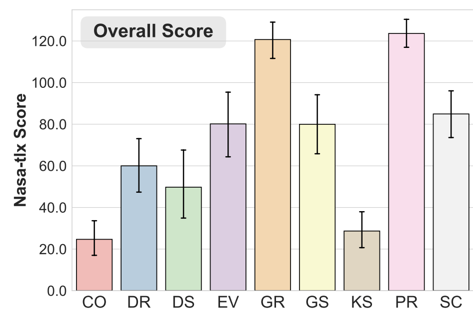

Fig. 4. Overall average NASA-TLX workload. Statistical significance of the six dimensions is reported in Section Sec. 5.5.3 and detailed in Appendix Sec. D.

For *Physical demand*, *Effort*, and *Frustration*, CloneObject performed best. Its physical demand was comparable to KeyboardSelect and significantly lower than DragSwitch ($p = .045$), ExplodedView ($p = .002$), and all remaining techniques ($p < .001$). Its effort was significantly lower than DragRotate ($p = .018$), ExplodedView ($p = .010$), and the other techniques ($p < .001$), with no difference from DragSwitch or KeyboardSelect. For frustration, it significantly outperformed ExplodedView ($p = .018$), PinchRotate ($p < .001$), and GestureRotate ($p < .001$). For *Performance*, KeyboardSelect scored best, significantly outperforming DragRotate ($p = .006$), ExplodedView ($p < .001$), PinchRotate ($p < .001$), and GestureRotate ($p < .001$), with no differences from the remaining techniques. For *Mental demand*, DragRotate scored best, significantly outperforming PinchRotate ($p = .001$), GestureRotate ($p = .040$), and SphereConnector ($p < .001$).

In summary, based on the mean NASA-TLX scores, CloneObject had the lowest overall workload and significantly outperformed all techniques except KeyboardSelect, including DragRotate ($p = .001$) and all others ($p < .001$).

5.5.4 UEQ-S

RM-ANOVA showed significant effects of Technique on *Pragmatic* ($F_{8,120} = 68.829,\ p < .001,\ \eta_p^2 = .821$), *Hedonic* ($F_{8,120} = 71.168,\ p < .001,\ \eta_p^2 = .992$), and *Overall* ($F_{8,120} = 78.799,\ p < .001,\ \eta_p^2 = .840$) UEQ-S scores.

From a usability (pragmatic) perspective, KeyboardSelect received the highest score. It did not differ from CloneObject

© 2026 IEEE. This is the author's version of the article accepted for publication in IEEE TVCG.
The final version is available at: https://doi.org/10.1109/TVCG.2026.3680697

or DragSwitch, but significantly outperformed ExplodedView ($p = .005$), GestureScale ($p = .004$), and all other techniques ($p < .001$). For hedonic quality, CloneObject performed best, with ExplodedView achieving a similar score. CloneObject significantly outperformed DragRotate ($p < .001$), GestureScale ($p = .032$), GestureRotate ($p < .001$), KeyboardSelect ($p < .001$), and PinchRotate ($p < .001$), with no differences from the remaining techniques.

For overall UEQ-S, CloneObject and DragSwitch had similar mean scores, both significantly higher than all other techniques. CloneObject did not differ from DragSwitch or ExplodedView, but significantly outperformed GestureScale ($p = .007$), KeyboardSelect ($p = .002$), and all others ($p < .001$). DragSwitch significantly outperformed GestureRotate ($p < .001$), KeyboardSelect ($p = .002$), PinchRotate ($p < .001$), and SphereConnector ($p < .001$), with no differences from the remaining techniques (see Tab. 1).

Technique	Pragmatic	Hedonic	Overall
CloneObject	5.34 (>avg.)	5.88 (exc.)	5.61 (exc.)
DragRotate	4.77 (>avg.)	2.75	3.76
DragSwitch	5.53 (>avg.)	5.66 (>avg.)	5.59 (>avg.)
ExplodedView	3.86	5.84 (>avg.)	4.85 (>avg.)
GestureRotate	2.42	2.78	2.60
GestureScale	4.38 (>avg.)	4.72 (>avg.)	4.55 (>avg.)
KeyboardSelect	5.62 (exc.)	3.77	4.70 (>avg.)
PinchRotate	2.08	1.89	1.98
SphereConnector	3.06	4.98 (>avg.)	4.02

Table 2. UEQ-S results (pragmatic, hedonic, overall). Yellow cells indicate values above the grand mean (">avg."), and pink cells indicate excellent performance ("exc.").

5.5.5 User Ranking

Fig. 5 shows participant rankings of the nine techniques. CloneObject and DragSwitch were each ranked most preferred by six participants (37.5% each). KeyboardSelect (assistive-tool metaphor) also received a relatively high mean rank (avg = 3.375). At the other end, nine participants (56.25%) ranked GestureRotate last and five (31.25%) ranked PinchRotate last; both use the eyeball-in-hand viewing metaphor. Participants reported that these techniques "*caused dizziness*" (P=13) and that "*the expected rotation direction did not match the actual rotation*" (P=11).

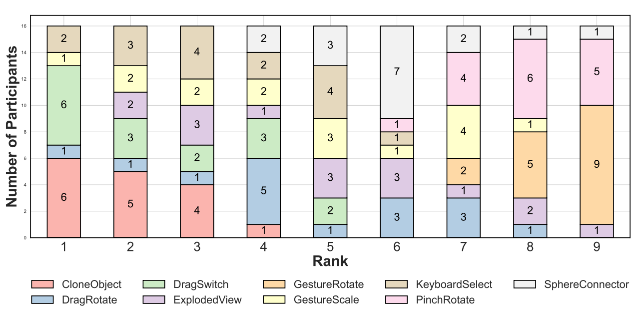

Fig. 5. Participant rankings of all techniques.

6 Discussion and Recommendations

Our results above show performance differences and user preferences among the techniques across scenarios defined by polyhedron radius (small vs. large) and face count (low vs. high). This section systematically discusses the strengths and weaknesses observed, followed by design recommendations to guide the design of future techniques.

6.1 Analysis by Scenario

Small-Radius Scenario: CloneObject significantly outperformed other techniques in selection time, likely due to its intuitive 6-DoF hand-following interaction, allowing users to rotate and inspect the target efficiently. Participants (N=7) described CloneObject as "*intuitive*" and "*effective*". Conversely, DragSwitch had significantly fewer errors. KeyboardSelect maintained balanced performance in both time and error rates. DragSwitch and KeyboardSelect's lower error rates are attributed to their discrete selection mechanisms, eliminating the instability associated with ray casting, which is consistent with prior findings. However, as P11 noted, KeyboardSelect *increased the time required to locate numbers on the keyboard*, which negatively affected the overall selection time. GestureRotate and GestureScale reduced error rates by magnifying the target face, yet participants (P12) highlighted increased selection times due to *repeated adjustments needed for optimal face positioning*.

Large-Radius Scenario: In large-radius conditions, where selection difficulty decreases, differences in selection time among techniques were less pronounced. CloneObject, DragRotate, and ExplodedView were efficient in terms of selection speed, whereas DragSwitch, GestureScale, and SphereConnector showed excellent error rates. CloneObject maintained superior performance due to its intuitive hand-following behavior. DragRotate, being the most familiar technique, *required minimal learning*, as participants (P6, P16) indicated. ExplodedView was controversial: although participants (P7, P12, P13) praised its *rapid identification of target faces*, the frequent occlusion *necessitated multiple adjustments*(P15), increasing error rates. SphereConnector effectively reduced ray-casting errors but had the highest mental demand according to NASA-TLX, as participants (N=4) indicated *difficulties in identifying corresponding sphere connections after object rotations*, thus prolonging selection time.

Low-Face-Count Scenario: CloneObject, ExplodedView, and KeyboardSelect demonstrated superior selection speed in low-face-count scenarios, whereas DragSwitch and KeyboardSelect excelled in accuracy. Although DragSwitch effectively reduced errors due to its discrete navigation, participants (N=5) mentioned difficulties in determining the next rotation step because of *limited visibility of adjacent faces*, resulting in increased selection times. KeyboardSelect, however, facilitated faster and more accurate selections as *fewer faces simplified key number localization on the keyboard*, commented by two participants (P11,P15).

High-Face-Count Scenario: For polyhedra with higher face counts, CloneObject showed significant time advantages, and DragSwitch excelled in error reduction. Despite CloneObject's speed, its error rate increased due to higher face density, which led to accidental adjacent-face selections. DragSwitch efficiently guided users by clearly indicating adjacent faces, reducing navigation time. KeyboardSelect, another discrete selection method, suffered increased errors and longer selection times due to the complexity of identifying numerical mappings on densely populated keyboards. Specifically, participant 8 encountered *confusion when distinguishing between numeric labels with similar shapes*, such as 6 and 9, due to their mirrored visual forms.

6.2 Evaluation by Design Dimensions

As stated in Sec. 3, our techniques are based on three dimensions: observation metaphor, disambiguation consistency, and selection metaphor.

Observation metaphor: Techniques employing "scene-in-hand" metaphors significantly outperformed "eyeball-in-hand" methods in both subjective and objective measures. Participants reported disorientation and unintended rotations with eyeball-in-hand interactions (PinchRotate and GestureRotate), reflected by their lowest NASA-TLX and UEQ-S usability scores.

Disambiguation consistency: ExplodedView, GestureScale, and GestureRotate altered spatial consistency during interaction; however, no participants reported significant cognitive confusion, possibly due to the preserved relative geometry of faces. Interestingly, some participants found these techniques enjoyable due to their enhanced 3D interactions.

Interaction metaphor: Auxiliary tool metaphors yielded the highest selection accuracy, outperforming virtual hand and ray-casting metaphors. Techniques employing auxiliary tools (DragSwitch and KeyboardSelect) received high NASA-TLX performance ratings, with users reporting stable and error-free interactions. Conversely, ray-casting techniques like ExplodedView, although highly engaging, suffered accuracy reductions due to occlusion issues. SphereConnector improved accuracy but increased cognitive load and selection time due to complex spatial reasoning.

6.3 Design Recommendations

Based on experimental results and user interviews, we provide the following practical guidelines:

- **DR1:** For scenarios demanding precision, particularly in complex conditions (small-radius or high-face-count), auxiliary tool-based interaction metaphors (e.g., DragSwitch and KeyboardSelect) are recommended, as these consistently achieved higher accuracy.
- **DR2:** "Eyeball-in-hand" observation metaphors (e.g., PinchRotate and GestureRotate) should be avoided for single-face selection tasks due to significantly lower usability and increased dizziness, as reported by participants across evaluation metrics.
- **DR3:** In entertainment-oriented applications, techniques altering disambiguation consistency (e.g., ExplodedView and GestureScale) are preferred. These enhance user enjoyment and foster engaging 3D interactions, as reflected by higher hedonic scores and positive participant feedback.

7 User Study 2: Application

In Study2, we examined whether the techniques identified in Study 1 on regular Polyhedra provide satisfactory interaction in more realistic and complex scenarios. To this end, we selected three techniques from Study 1 that demonstrated representative strengths across different scenarios and evaluated their usability for face selection on objects with varying geometric complexity: a drum, a cup, and a gear. For these objects, selectable regions were defined as generalized, salient parts rather than strict mesh faces, reflecting how users typically perceive and interact with real-world items.

7.1 Participants and Apparatus

We recruited another 16 participants from a local university (9 female, 7 male) with prior experience in 3D modeling (e.g., Maya, Blender), primarily from architecture and digital media programs. Participants were 20–29 years old ($M = 23.4$, $SD = 2.6$), and all were right-handed. All had normal or corrected-to-normal vision with no reported color-vision deficiencies. None had participated in Study 1. Self-reported VR familiarity on a 5-point scale (1 = not at all familiar, 5 = very familiar) was moderate ($M = 2.34$, $SD = 0.932$). The experiment was conducted in a virtual environment using a Meta Quest 3 with the same system setup as in Study 1 (Sec. 5).

7.2 Interaction Object Design and Procedure

Participants were instructed to use the three techniques recommended in User Study 1 (KeyboardSelect, CloneObject, and DragSwitch) to select a single face on each of the three complex objects, as shown in Fig. 6. To ensure that the evaluation encompassed scenarios commonly encountered in real-world applications, we selected three objects that featured curved surfaces with a high density of faces and intricate structural configurations. The object shown in Fig. 6a is a cup, which includes three selectable faces, two of which are curved surfaces (i.e., the handle and the top part of the cup). The object in Fig. 6c is a gear composed of 49 rectangular planes and 2 irregular faces, and while the height of each rectangular face is consistent, their widths vary. The object in Fig. 6b is a drum characterized by a highly complex structure. For this object, we selected a variety of face types, including two large circular planes, fifteen smaller rectangular planes, and sixteen curved surfaces.

The implementations of KeyboardSelect and CloneObject were consistent with those used in Study 1. However, the red bounding box used in DragSwitch proved less effective for curved surfaces. To address this limitation, we replaced the bounding box with a yellow highlight when curved surfaces were involved. In each trial, one randomly chosen face was highlighted in red to indicate the target face, and participants were required to select this face using all three techniques.

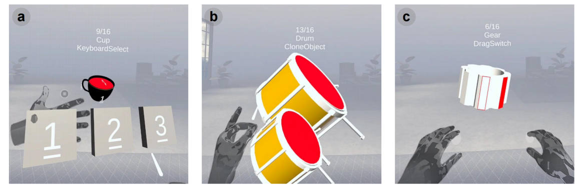

Fig. 6. Representative examples of the techniques and objects used in the study. Participants applied each technique to select faces on each of the three complex polyhedral objects. Shown are: KeyboardSelect on Cup (left), CloneObject on Drum (center), and DragSwitch on Gear (right).

This experiment followed a similar procedure to Study 1. Participants first completed a demographic questionnaire, then received an introduction to the experiment Sec. 5.4, and finally participated in three experimental sessions, one for each technique, presented in a randomized order. In each session, participants first learned how to use the given technique and then performed 10 selection tasks on polyhedra under different conditions using that technique. At the end of each session, they completed a short version of UEQ-S and did a semi-structured interview. The entire experiment lasted about 30 minutes.

7.3 Results

The results of the UEQ-S are summarized in Tab. 2. The data indicate that when applied to single-face selection on more complex polyhedra, CloneObject and DragSwitch outperformed KeyboardSelect in pragmatic, hedonic, and overall attractiveness. Next, we provide a summary of interview responses grouped by technique.

Technique	Pragmatic	Hedonic	Overall
CloneObject	2.69(>exc.)	2.31(>avg.)	2.50(>exc.)
DragSwitch	2.38(>avg.)	2.44(>exc.)	2.41(>avg.)
KeyboardSelect	1.69	1.50	1.59

Table 3. Results from the UEQ-S, showing the pragmatic quality, hedonic quality, and overall quality of each recommended technique.

7.3.1 CloneObject

The CloneObject technique generates a smaller-scale duplicate of the original object, allowing users to observe the target face from different angles by rotating their hand. Most participants (N=9) perceived this technique as the most "*efficient*" and "*effort-saving*", with several participants (N=6) highlighting its strong applicability and noting that "*this technique provided a consistently positive experience across different complex polyhedral selections*" (P12).

However, some participants (N=5) reported challenges when attempting to select faces located on the back side of the object. One participant noted, "*When selecting a face on the back of the shape, my right hand sometimes blocks my view of the object, making it difficult to determine whether the correct face has been selected*" (P2). Another participant commented, "*When the target face is on the backside, my left hand needs to rotate at a large angle to locate it*" (P8). Furthermore, most participants (N=11) expressed concerns regarding face selection accuracy when interacting with smaller surfaces, such as the small rectangular faces on the drum, stating that the proximity of neighboring faces "*increased the likelihood of accidental selection*"(P4).

© 2026 IEEE. This is the author's version of the article accepted for publication in IEEE TVCG.
The final version is available at: https://doi.org/10.1109/TVCG.2026.3680697

7.3.2 DragSwitch

Half of the participants (N=8) noted that *this technique enabled more accurate selection when targeting smaller surfaces*. However, several participants (N=6) pointed out that *the technique required frequent switching when dealing with objects containing a large number of faces*. One participant commented, "*If the initial position of the red frame is far from the target face, multiple swipes are needed to reach it*" (P11), while another remarked, "*This technique can become tiring if frequent switching is required*" (P5).

7.3.3 KeyboardSelect

Most of the participants (N = 8) reported that this technique was *more effective when the object contained a small number of faces*, noting its convenience, particularly in the cup condition.

However, participants also highlighted its limitations when applied to more complex structures. One participant remarked, "*When the object is a non-convex polyhedron, the numbers on recessed faces (where the angle between the face normal and those of adjacent faces exceeds 180 degrees) are sometimes occluded by neighboring faces*" (P1). Other participants pointed out challenges related to face identification as the number of faces increased, "*When there are many faces, the keyboard in hand displays too many numbers, making it time-consuming to locate the correct one*" (P7). Similarly, as another participant noted, "*When the face density is high, it requires more cognitive effort to match each number with its corresponding face*" (P15).

Additionally, some participants (N=4) expressed concern that "*the presence of visible numbering feels somewhat intrusive*" and "*may negatively impact the immersive experience in VR*".

7.4 Discussion

In this section, we discuss and summarize the results and propose solutions and design implications given each technique's limitations.

CloneObject is a broadly applicable and practical technique. It received the highest scores for both pragmatic quality and overall satisfaction in the UEQ-S. However, CloneObject may be prone to selection errors due to hand tremors when targeting smaller areas. One potential solution is inspired by the concept of TinyHand [4], which suggests dynamically reducing the hand-to-object mapping scale when the hand approaches smaller surfaces, thereby enabling more precise interaction. Additionally, user interviews revealed challenges in selecting faces located on the back side of the object. This issue arises because the cloned object always maintains the same orientation as the original, allowing users to view and interact primarily with the front-facing side, while the rear remains difficult to access. A possible workaround is to rotate the object to face the target area before cloning. However, implementing this solution in bare-hand interaction environments is non-trivial, as the available delimiters or triggers are limited. This calls for more refined gesture designs to facilitate such functionality effectively.

DragSwitch employs a discrete selection mechanism, which can offer higher selection accuracy. This technique is relatively novel and received the highest hedonic quality score in the UEQ-S questionnaire. However, it also has certain limitations. First, when applied to shapes with curved surfaces, the red bounding box cannot accurately outline curved edges, necessitating the use of alternative highlighting methods. Moreover, the red frame can only move to the nearest face in the direction of movement. When multiple faces lie between the current and target face, moving the frame to the target may require significant physical effort and time.

Similarly, KeyboardSelect employs a discrete selection mechanism to improve accuracy. However, its applicability is limited, and it performs poorly when used with real-world polyhedral models, receiving the lowest score on the UEQ-S questionnaire. In non-convex 3D shapes, numeric labels are often occluded by adjacent surfaces. On small faces, such labels may either overflow beyond the boundaries or become too small to be clearly legible. Moreover, the persistent presence of visible numbers detracts from the immersive experience in VR, particularly when interacting with complex polyhedral structures.

8 Limitations and Future Work

We identified four limitations that can serve as future exploration lines. First, no single technique is suitable for all scenarios, and our research provides recommendations on which technique to use in different scenarios. We plan to explore the design of a universal technique that can be applied to multiple contexts. Second, this study does not examine the long-term use of each technique. As the number of selectable faces increases, keyboard-assisted selection showed reduced scalability due to higher visual search effort. Hierarchical numerical input (e.g., multi-step selection) is a promising direction for future work. Third, our application only discusses polyhedra unfoldings that may occur in three practical applications. Fourth, we did not test these designed techniques in conjunction with more complex sculpting and modeling tools/features. Our work focuses on generalized faces at the level of user understanding; extending these techniques to densely arranged graphics-level mesh faces in high-poly models remains an important direction for future work. Further research could expand gesture input patterns to accommodate more advanced operational features that are part of the workflow process for such tasks. Fifth, our current techniques do not leverage fine-grained finger tracking or interaction modalities that are strongly coupled with detailed hand input. Future work could investigate how the integration of such advanced hand tracking technologies might further improve selection accuracy, efficiency, and user experience in complex modeling tasks. Sixth, our participant sample was primarily drawn from university students majoring in computer science and microelectronics, resulting in a higher proportion of VR-experienced users. Broader recruitment will be considered in future studies to improve generalizability. In addition, this work provides an initial exploration of gesture-based techniques for face-level selection. Investigating how the strengths of these techniques can be combined represents an important direction for future work.

9 Conclusion

In this study, we designed and evaluated techniques for selecting elements within a larger 3D structure, such as the faces of a polyhedron in VR. Specifically, we constructed a design space consisting of three dimensions: viewing strategy, disambiguation consistency, and interaction metaphor. Based on this design space, we developed eight new techniques on top of traditional rotational observation and ray-based DragRotate techniques, and evaluated them through the first user study. The results allowed us to analyze which techniques were suitable for different scenarios, explore the impact of the three design dimensions on the outcomes, and identify three top-performing techniques (CloneObject, DragSwitch, and KeyboardSelect). Building on this, we conducted a second study to analyze the applicability of the three techniques in more realistic single-face selection tasks. Overall, our work on the design space we conceptualized and the techniques we developed can inspire exciting user interface and application designs related to the selection of elemental parts of larger 3D shapes.

References

[1] F. Argelaguet and C. Andujar. Efficient 3d pointing selection in cluttered virtual environments. *IEEE Comput. Graph. Appl.*, 29(6):34–43, nov 2009. 2

[2] F. Argelaguet, C. Andujar, and R. Trueba. Overcoming eye-hand visibility mismatch in 3d pointing selection. In *Proceedings of the 2008 ACM Symposium on Virtual Reality Software and Technology*, VRST '08, p. 43–46. Association for Computing Machinery, New York, NY, USA, 2008. doi: 10.1145/1450579.1450588 2

[3] Autodesk Inc. Maya 2022 user guide: Selecting components, 2022. Accessed: 2024-04-03. 2

[4] S. Bhowmick, N. Biswas, P. C. Kalita, and K. Sorathia. Wow! I have tiny hands: Design and evaluation of adaptive virtual hands for small object selection within arms length in dense virtual environment. In *Extended Abstracts of the 2023 CHI Conference on Human Factors in Computing Systems*, CHI EA '23. Association for Computing Machinery, New York, NY, USA, 2023. doi: 10.1145/3544549.3585686 1, 2, 9

[5] Blender Foundation. Blender manual: Selecting meshes, 2024. Accessed: 2024-04-03. 2

[6] P. Boffi, A. Kouyoumdjian, M. Waldner, P. L. Lanzi, and I. Viola. Bagginghook: Selecting moving targets by pruning distractors away for intention-prediction heuristics in dense 3d environments. In *2024 IEEE Conference Virtual Reality and 3D User Interfaces (VR)*, pp. 913–923, 2024. doi: 10.1109/VR58804.2024.00110 2

[7] D. Bowman, C. Wingrave, J. Campbell, and V. Ly. Using pinch gloves for both natural and abstract interaction techniques in virtual environments. *Proceedings of HCI International*, 01 2001. 2, 3

[8] D. Bowman, C. Wingrave, J. Campbell, V. Ly, and C. Rhoton. Novel uses of pinch gloves™ for virtual environment interaction techniques. *Virtual Reality*, 6:122–129, 10 2002. doi: 10.1007/s100550200013 2, 3

[9] D. A. Bowman, R. P. McMahan, and E. D. Ragan. Questioning naturalism in 3d user interfaces. *Commun. ACM*, 55(9):78–88, Sept. 2012. doi: 10.1145/2330667.2330687 2

[10] G. Buckingham. Hand tracking for immersive virtual reality: Opportunities and challenges. *Frontiers in Virtual Reality*, 2, 2021. doi: 10.3389/frvir.2021.728461 2

[11] S. Büttner, M. Prilla, and C. Röcker. Augmented reality training for industrial assembly work - are projection-based ar assistive systems an appropriate tool for assembly training? In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, CHI '20, p. 1–12. Association for Computing Machinery, New York, NY, USA, 2020. doi: 10.1145/3313831.3376720 1

[12] J. Cashion, C. Wingrave, and J. J. LaViola Jr. Dense and dynamic 3d selection for game-based virtual environments. *IEEE Transactions on Visualization and Computer Graphics*, 18(4):634–642, Apr. 2012. doi: 10.1109/TVCG.2012.40 2, 3

[13] Y. Chen, J. Sun, Q. Xu, E. Lank, P. Irani, and W. Li. Empirical evaluation of moving target selection in virtual reality using egocentric metaphors. In C. Ardito, R. Lanzilotti, A. Malizia, H. Petrie, A. Piccinno, G. Desolda, and K. Inkpen, eds., *Human-Computer Interaction – INTERACT 2021*, pp. 29–50. Springer International Publishing, Cham, 2021. 2

[14] K. Danyluk and W. Willett. Evaluating the performance of virtual reality navigation techniques for large environments. In M. Gavrilova, J. Chang, N. M. Thalmann, E. Hitzer, and H. Ishikawa, eds., *Advances in Computer Graphics*, pp. 203–215. Springer International Publishing, Cham, 2019. 3

[15] A. David, E. Joy, S. Kumar, and S. J. Bezaleel. Integrating virtual reality with 3d modeling for interactive architectural visualization and photorealistic simulation: A direction for future smart construction design using a game engine. In J. I.-Z. Chen, J. M. R. S. Tavares, A. M. Iliyasu, and K.-L. Du, eds., *Second International Conference on Image Processing and Capsule Networks*, pp. 180–192. Springer International Publishing, Cham, 2022. 3

[16] W. Delamare, M. Daniel, and K. Hasan. Multifingerbubble: A 3d bubble cursor variation for dense environments. In *Extended Abstracts of the 2022 CHI Conference on Human Factors in Computing Systems*, CHI EA '22. Association for Computing Machinery, New York, NY, USA, 2022. doi: 10.1145/3491101.3519692 1, 2, 3

[17] T. J. Dube and A. S. Arif. Text entry in virtual reality: A comprehensive review of the literature. In *Human-Computer Interaction. Recognition and Interaction Technologies: Thematic Area, HCI 2019, Held as Part of the 21st HCI International Conference, HCII 2019, Orlando, FL, USA, July 26–31, 2019, Proceedings, Part II 21*, pp. 419–437. Springer, 2019. 3

[18] N. Elmqvist and P. Tsigas. A taxonomy of 3d occlusion management techniques. In *2007 IEEE Virtual Reality Conference*, pp. 51–58, 2007. doi: 10.1109/VR.2007.352463 1, 2

[19] J. W. Filho, W. Stuerzlinger, and L. Nedel. Comparing and combining virtual hand and virtual ray pointer interactions for data manipulation in immersive analytics. *Transactions on Visualization and Computer Graphics (TVCG)*, 27(5):2513–2523, 2021. VR '21. doi: 10.1109/TVCG.2021.3067759 2

[20] A. Forsberg, K. Herndon, and R. Zeleznik. Aperture based selection for immersive virtual environments. In *Proceedings of the 9th Annual ACM Symposium on User Interface Software and Technology*, UIST '96, p. 95–96. Association for Computing Machinery, New York, NY, USA, 1996. doi: 10.1145/237091.237105 2

[21] S. Frees and G. Kessler. Precise and rapid interaction through scaled manipulation in immersive virtual environments. In *IEEE Proceedings. VR 2005. Virtual Reality, 2005.*, pp. 99–106, 2005. doi: 10.1109/VR.2005.1492759 2

[22] S. Frees, G. D. Kessler, and E. Kay. Prism interaction for enhancing control in immersive virtual environments. *ACM Trans. Comput.-Hum. Interact.*, 14(1):2–es, May 2007. doi: 10.1145/1229855.1229857 2

[23] E. Godden, W. Steedman, and M. K. Pan. Robotic characterization of markerless hand-tracking on meta quest pro and quest 3 virtual reality headsets. *IEEE Transactions on Visualization and Computer Graphics*, 31(5):3025–3034, 2025. doi: 10.1109/TVCG.2025.3549182 5

[24] T. Grossman and R. Balakrishnan. The bubble cursor: enhancing target acquisition by dynamic resizing of the cursor's activation area. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '05, p. 281–290. Association for Computing Machinery, New York, NY, USA, 2005. doi: 10.1145/1054972.1055012 1

[25] D. Han, D. Kim, and I. Cho. Portal: Portal widget for remote target acquisition and control in immersive virtual environments. In *Proceedings of the 28th ACM Symposium on Virtual Reality Software and Technology*, VRST '22. Association for Computing Machinery, New York, NY, USA, 2022. doi: 10.1145/3562939.3565639 2

[26] S. G. Hart. Nasa-task load index (nasa-tlx); 20 years later. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, 50(9):904–908, 2006. doi: 10.1177/154193120605000909 5

[27] X. Hu, W. Xu, Y. Wei, H. Zhang, J. Huang, and H.-N. Liang. Optimizing moving target selection in vr by integrating proximity-based feedback types and modalities. In *2025 IEEE Conference Virtual Reality and 3D User Interfaces (VR)*, pp. 52–62, 2025. doi: 10.1109/VR59515.2025.00030 6

[28] X. Hu, X. Yan, Y. Wei, W. Xu, Y. Li, Y. Liu, and H.-N. Liang. Exploring the effects of spatial constraints and curvature for 3d piloting in virtual environments. In *2024 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pp. 505–514, 2024. doi: 10.1109/ISMAR62088.2024.00065 2

[29] X. Hu, Y. Zhang, Y. Wei, L. Zhang, Y. Li, W. Stuerzlinger, and H.-N. Liang. Exploring and modeling the effects of eye-tracking accuracy and precision on gaze-based steering in virtual environments. *IEEE Transactions on Visualization and Computer Graphics*, 31(11):9559–9569, 2025. doi: 10.1109/TVCG.2025.3616824 2

[30] J. Huang, J. J. Dudley, S. Uzor, D. Wu, P. O. Kristensson, and F. Tian. Understanding user performance of acquiring targets with motion-in-depth in virtual reality. *International Journal of Human-Computer Studies*, 163:102817, 2022. 2

[31] J. Huang, F. Tian, X. Fan, H. Tu, H. Zhang, X. Peng, and H. Wang. Modeling the endpoint uncertainty in crossing-based moving target selection. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, CHI '20, p. 1–12. Association for Computing Machinery, New York, NY, USA, 2020. doi: 10.1145/3313831.3376336 2

[32] J. Huang, F. Tian, X. Fan, X. L. Zhang, and S. Zhai. Understanding the uncertainty in 1d unidirectional moving target selection. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, CHI '18, p. 1–12. Association for Computing Machinery, New York, NY, USA, 2018. doi: 10.1145/3173574.3173811 2

[33] J. Huang, F. Tian, N. Li, and X. Fan. Modeling the uncertainty in 2d moving target selection. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology*, UIST '19, p. 1031–1043. Association for Computing Machinery, New York, NY, USA, 2019. doi: 10.1145/3332165.3347880 2

[34] B. Jackson, K. Beckham, A. K. Cohen, and B. C. Heggeseth. Comparing convex region-of-interest selection techniques for surface geometry. In *Proceedings of the 25th ACM Symposium on Virtual Reality Software and Technology*, VRST '19. Association for Computing Machinery, New York, NY, USA, 2019. doi: 10.1145/3359996.3364258 2

[35] S. K. Jagatheesaperumal, P. Sathikumar, and H. Rajan. Metadigihuman: Haptic interfaces for digital humans in metaverse, 2024. 2

[36] P. Kabbash and W. A. S. Buxton. The "prince" technique: Fitts' law and selection using area cursors. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '95, p. 273–279. ACM Press/Addison-Wesley Publishing Co., USA, 1995. doi: 10.1145/223904.223939 2

[37] D. Kong, H. Wang, S. Zhou, H. Cui, and Z. Feng. Penlab: Towards understanding of active collaboration for solid geometry teaching. In *Extended Abstracts of the 2024 CHI Conference on Human Factors in Computing Systems*, CHI EA '24. Association for Computing Machinery, New York, NY, USA, 2024. doi: 10.1145/3613905.3650789 3

[38] M. Krüger, T. Gerrits, T. Römer, T. Kuhlen, and T. Weissker. Intenselect+: Enhancing score-based selection in virtual reality. *IEEE Transactions on Visualization and Computer Graphics*, 30(5):2829–2838, 2024. doi: 10.

© 2026 IEEE. This is the author's version of the article accepted for publication in IEEE TVCG.
The final version is available at: https://doi.org/10.1109/TVCG.2026.3680697

1109/TVCG.2024.3372077 2

[39] Y. Li, J. Liu, J. Huang, Y. Zhang, X. Peng, Y. Bian, and F. Tian. Evaluating the effects of user motion and viewing mode on target selection in augmented reality. *International Journal of Human-Computer Studies*, 191:103327, 2024. 2

[40] J. Liang and M. Green. Jdcad: A highly interactive 3d modeling system. *Computers & Graphics*, 18(4):499–506, 1994. doi: 10.1016/0097-8493(94)90062-0 2

[41] L. Luo, P. N. Chowdhury, T. Xiang, Y.-Z. Song, and Y. Gryaditskaya. 3d vr sketch guided 3d shape prototyping and exploration, 2024. 3

[42] L. Luo, Y. Gryaditskaya, Y. Yang, T. Xiang, and Y.-Z. Song. Towards 3d vr-sketch to 3d shape retrieval. In *2020 International Conference on 3D Vision (3DV)*, pp. 81–90, 2020. doi: 10.1109/3DV50981.2020.00018 3

[43] K. Mania, D. Wooldridge, M. Coxon, and A. Robinson. The effect of visual and interaction fidelity on spatial cognition in immersive virtual environments. *IEEE Transactions on Visualization and Computer Graphics*, 12(3):396–404, 2006. doi: 10.1109/TVCG.2006.55 3

[44] D. Mendes, F. M. Caputo, A. Giachetti, A. Ferreira, and J. Jorge. A survey on 3d virtual object manipulation: From the desktop to immersive virtual environments. *Computer Graphics Forum*, 38(1):21–45, 2019. doi: 10.1111/cgf.13390 2

[45] M. R. Mine. Virtual environment interaction techniques. *UNC Chapel Hill CS Dept*, 1995. 3

[46] M. R. Mine, F. P. Brooks, and C. H. Sequin. Moving objects in space: exploiting proprioception in virtual-environment interaction. In *Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '97, p. 19–26. ACM Press/Addison-Wesley Publishing Co., USA, 1997. doi: 10.1145/258734.258747 2, 3

[47] A. Pavlovych and C. Gutwin. Assessing target acquisition and tracking performance for complex moving targets in the presence of latency and jitter. In *Proceedings of Graphics Interface 2012*, GI '12, p. 109–116. Canadian Information Processing Society, CAN, 2012. 2

[48] D.-M. Pham and W. Stuerzlinger. Hawkey: Efficient and versatile text entry for virtual reality. In *Proceedings of the 25th ACM Symposium on Virtual Reality Software and Technology*, VRST '19. Association for Computing Machinery, New York, NY, USA, 2019. doi: 10.1145/3359996.3364265 2

[49] I. Poupyrev, M. Billinghurst, S. Weghorst, and T. Ichikawa. The go-go interaction technique: non-linear mapping for direct manipulation in vr. In *Proceedings of the 9th Annual ACM Symposium on User Interface Software and Technology*, UIST '96, p. 79–80. Association for Computing Machinery, New York, NY, USA, 1996. doi: 10.1145/237091.237102 2

[50] G. Ren and E. O'Neill. 3d selection with freehand gesture. *Computers & Graphics*, 37(3):101–120, 2013. doi: 10.1016/j.cag.2012.12.006 2

[51] A. Renaux, F. Muhla, F. Clanché, P. Meyer, S. Maïaux, S. Colnat-Coulbois, and G. Gauchard. Effects of using immersive virtual reality on time and steps during a locomotor task in young adults. *PLOS ONE*, 17(10):1–13, 10 2022. doi: 10.1371/journal.pone.0275876 2

[52] W. Robinett and R. Holloway. Implementation of flying, scaling and grabbing in virtual worlds. In *Proceedings of the 1992 Symposium on Interactive 3D Graphics*, I3D '92, p. 189–192. Association for Computing Machinery, New York, NY, USA, 1992. doi: 10.1145/147156.147201 2

[53] M. Schrepp, J. Thomaschewski, and A. Hinderks. Design and Evaluation of a Short Version of the User Experience Questionnaire (UEQ-S). *International Journal of Interactive Multimedia and Artificial Intelligence*, 4(6):103–108, Dec. 2017. doi: 10.9781/ijimai.2017.09.001 5

[54] R. Shi, Y. Wei, X. Hu, Y. Liu, Y. Yue, L. Yu, and H.-N. Liang. Experimental analysis of freehand multi-object selection techniques in virtual reality head-mounted displays. *Proc. ACM Hum.-Comput. Interact.*, 8(ISS), Oct. 2024. doi: 10.1145/3698129 3

[55] O. Soto-Martín, A. Fuentes-Porto, and J. Martín-Gutiérrez. A digital reconstruction of a historical building and virtual reintegration of mural paintings to create an interactive and immersive experience in virtual reality. *Applied Sciences*, 2020. 3

[56] H. Tu, J. Huang, H.-N. Liang, R. Skarbez, F. Tian, and H. B.-L. Duh. Distractor effects on crossing-based interaction. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, pp. 1–13, 2021. 2

[57] Y. Wang, Y. Li, and H.-N. Liang. Comparative analysis of artefact interaction and manipulation techniques in vr museums: A study of performance and user experience. In *2023 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pp. 761–770, 2023. doi: 10.1109/ISMAR59233.2023.00091 3

[58] C. Ware and S. Osborne. Exploration and virtual camera control in virtual three dimensional environments. In *Proceedings of the 1990 Symposium on Interactive 3D Graphics*, I3D '90, p. 175–183. Association for Computing Machinery, New York, NY, USA, 1990. doi: 10.1145/91385.91442 3

[59] J. Wei, K. Zhu, and Y. Ye. Fanray: Using fan mechanism in ray-casting to improve target selection in vr. In *Proceedings of the 2023 3rd International Conference on Human Machine Interaction*, ICHMI '23, p. 1–6. Association for Computing Machinery, New York, NY, USA, 2023. doi: 10.1145/3604383.3604395 1, 2

[60] R. Weller, W. Wegele, C. Schröder, and G. Zachmann. Lenselect: Object selection in virtual environments by dynamic object scaling. In *Frontiers in Virtual Reality*, 2021. 1, 2

[61] N. Wenk, J. Penalver-Andres, K. Buetler, et al. Effect of immersive visualization technologies on cognitive load, motivation, usability, and embodiment. *Virtual Reality*, 27:307–331, 2023. doi: 10.1007/s10055-021-00565-8 3

[62] Y. Wu, Y. Wang, X. Lou, et al. An empirical practice of design and evaluation of freehand interaction gestures in virtual reality. *Multimedia Tools and Applications*, 83:52481–52507, 2024. doi: 10.1007/s11042-023-17640-8 2

[63] Z. Wu, D. Yu, and J. Goncalves. Point- and volume-based multi-object acquisition in vr. In *Human-Computer Interaction – INTERACT 2023: 19th IFIP TC13 International Conference, York, UK, August 28 – September 1, 2023, Proceedings, Part I*, p. 20–42. Springer-Verlag, Berlin, Heidelberg, 2023. doi: 10.1007/978-3-031-42280-5_2 2

[64] D. Yu, T. Dingler, E. Velloso, and J. Goncalves. Object selection and manipulation in vr headsets: Research challenges, solutions, and success measurements. *ACM Comput. Surv.*, 57(4), Dec. 2024. doi: 10.1145/3706417 2

[65] D. Yu, T. Dingler, E. Velloso, and J. Goncalves. Object selection and manipulation in vr headsets: Research challenges, solutions, and success measurements. *ACM Comput. Surv.*, 57(4), Dec. 2024. doi: 10.1145/3706417 3

[66] D. Yu, X. Lu, R. Shi, H.-N. Liang, T. Dingler, E. Velloso, and J. Goncalves. Gaze-supported 3d object manipulation in virtual reality. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, CHI '21. Association for Computing Machinery, New York, NY, USA, 2021. doi: 10.1145/3411764.3445343 6

[67] D. Yu, Q. Zhou, T. Dingler, E. Velloso, and J. Gonçalves. Blending on-body and mid-air interaction in virtual reality. *2022 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pp. 637–646, 2022. 2

[68] D. Yu, Q. Zhou, J. Newn, T. Dingler, E. Velloso, and J. Goncalves. Fully-occluded target selection in virtual reality. *IEEE Transactions on Visualization and Computer Graphics*, 26(12):3402–3413, 2020. PMID: 32986552. doi: 10.1109/TVCG.2020.3023606 1, 2, 3, 4

[69] E. Yu, F. Chevalier, K. Singh, and A. Bousseau. 3d-layers: Bringing layer-based color editing to vr painting. *ACM Trans. Graph.*, 43(4), July 2024. doi: 10.1145/3658183 1

[70] Y. Zheng, J. Huang, H. Zhang, Y. Bian, J. Liu, C. Yang, F. Tian, and X. Meng. 3d ternary-gaussian model: Modeling pointing uncertainty of 3d moving target selection in virtual reality. *International Journal of Human-Computer Studies*, 198:103454, 2025. 2

[71] X. Zhou, H. Qin, W. Xiao, L. Jia, and C. Xue. A comparative usability study of bare hand three-dimensional object selection techniques in virtual environment. *Symmetry*, 12(10), 2020. 2

[72] F. Zoccheddu, E. Gobbetti, M. Livesu, N. Pietroni, and G. Cherchi. Hexbox: interactive box modeling of hexahedral meshes. In *Computer Graphics Forum*, vol. 42, p. e14899. Wiley Online Library, 2023. 1, 3

A Code Availability

All scripts for the interaction techniques and experimental setup will be made available via GitHub after the review process is finalized.

B Supplementary Material

We conducted six sets of pilot studies and the results showed significant differences in time $(F_{1,5} = 121.634,\ p < .001)$ and error rate $(F_{1,5} = 72.265,\ p < .001)$ between our small-radius and large-radius conditions, as illustrated in Fig. 7.

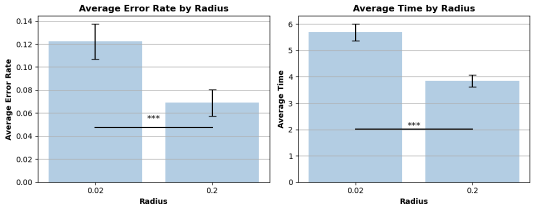

Fig. 7. Comparison of task completion time (seconds) and error rate (%) between small-radius and large-radius conditions. Results show significant differences between the two conditions ($p < 0.001$).

C Statistical Test Results

This appendix presents pairwise significance results under different experimental conditions. Specifically, the results are categorized based on radius size (small vs. large) and face count (less vs. more). The significance values are organized in matrix format for clarity.

Tables 4 and 5 present pairwise p-values for selection time under small and large radius conditions. Tables 8 and 9 present pairwise p-values for selection time under less and more face conditions. Tables 6 and 7 present pairwise p-values for error rate under small and large radius conditions. Tables 10 and 11 present pairwise p-values for error rate under less and more face conditions.

The following abbreviations are used: CO = *CloneObject*, DR = *DragRotate*, DS = *DragSwitch*, EV = *ExplodedView*, GR = *GestureRotate*, GS = *GestureScale*, KS = *KeyboardSelect*, PR = *PinchRotate*, SC = *SphereConnector*.

	CO	DR	DS	EV	GR	GS	KS	PR	SC
CO	–	0.030	0.001	0.050	< .001	< .001	0.001	< .001	< .001
DR		–	1.000	1.000	0.092	1.000	0.471	0.046	1.000
DS			–	1.000	0.001	< .001	1.000	0.001	0.009
EV				–	< .001	0.443	1.000	< .001	1.000
GR					–	0.106	< .001	1.000	0.456
GS						–	< .001	0.045	1.000
KS							–	< .001	0.006
PR								–	0.041
SC									–

Table 4. Pairwise p-values for selection time under small radius condition. Values are shown only in the lower triangle for clarity.

	CO	DR	DS	EV	GR	GS	KS	PR	SC
CO	–	1.000	0.015	1.000	0.012	< .001	0.029	0.009	< .001
DR		–	1.000	1.000	0.009	< .001	0.004	0.090	0.112
DS			–	1.000	0.144	0.011	1.000	1.000	1.000
EV				–	0.003	< .001	0.411	0.255	0.388
GR					–	1.000	0.094	0.048	0.207
GS						–	0.053	0.002	0.088
KS							–	1.000	1.000
PR								–	1.000
SC									–

Table 5. Pairwise p-values for selection time under large radius condition. Values are shown only in the upper triangle for clarity.

D NASA-TLX Subscale Scores

The distribution of NASA-TLX sub-scale scores for all techniques is illustrated in Figure 8.

	CO	DR	DS	EV	GR	GS	KS	PR	SC
CO	–	0.004	0.131	0.011	1.000	1.000	0.228	< .001	1.000
DR		–	< .001	1.000	0.001	< .001	< .001	0.351	< .001
DS			–	< .001	0.885	0.936	1.000	< .001	0.202
EV				–	< .001	0.001	< .001	1.000	< .001
GR					–	1.000	1.000	< .001	1.000
GS						–	1.000	< .001	1.000
KS							–	< .001	0.422
PR								–	< .001
SC									–

Table 6. Pairwise p-values for error rate under small radius condition. Values are shown only in the upper triangle for clarity.

	CO	DR	DS	EV	GR	GS	KS	PR	SC
CO	–	1.000	0.067	0.070	1.000	0.284	1.000	1.000	0.057
DR		–	1.000	0.052	1.000	1.000	1.000	1.000	1.000
DS			–	< .001	0.645	1.000	1.000	0.058	1.000
EV				–	0.021	0.001	< .001	0.134	0.002
GR					–	1.000	1.000	1.000	1.000
GS						–	1.000	0.068	1.000
KS							–	0.651	1.000
PR								–	0.284
SC									–

Table 7. Pairwise p-values for error rate under large radius condition. Values are shown only in the upper triangle for clarity.

	CO	DR	DS	EV	GR	GS	KS	PR	SC
CO	–	0.018	0.019	1.000	< .001	< .001	0.019	< .001	< .001
DR		–	1.000	1.000	0.001	0.003	0.671	0.006	0.089
DS			–	1.000	0.285	1.000	1.000	1.000	1.000
EV				–	< .001	< .001	1.000	< .001	0.161
GR					–	1.000	< .001	1.000	1.000
GS						–	< .001	1.000	1.000
KS							–	0.001	0.007
PR								–	1.000
SC									–

Table 8. Pairwise p-values for selection time under less faces condition. Values are shown only in the upper triangle for clarity.

	CO	DR	DS	EV	GR	GS	KS	PR	SC
CO	–	0.024	0.004	0.042	< .001	< .001	0.003	< .001	0.001
DR		–	0.408	1.000	< .001	0.858	1.000	0.030	1.000
DS			–	1.000	< .001	< .001	0.142	0.001	0.005
EV				–	< .001	0.482	1.000	0.013	1.000
GR					–	0.137	0.005	1.000	0.014
GS						–	0.084	1.000	1.000
KS							–	0.029	1.000
PR								–	0.151
SC									–

Table 9. Pairwise p-values for selection time under more faces condition. Values are shown only in the upper triangle for clarity.

	CO	DR	DS	EV	GR	GS	KS	PR	SC
CO	–	1.000	1.000	0.279	1.000	1.000	1.000	< .001	1.000
DR		–	0.104	1.000	1.000	0.155	0.139	0.355	1.000
DS			–	0.014	1.000	1.000	1.000	< .001	1.000
EV				–	0.012	0.010	0.001	1.000	0.087
GR					–	1.000	1.000	< .001	1.000
GS						–	1.000	< .001	1.000
KS							–	< .001	0.900
PR								–	< .001
SC									–

Table 10. Pairwise p-values for error rate under less faces condition. Values are shown only in the upper triangle for clarity.

	CO	DR	DS	EV	GR	GS	KS	PR	SC
CO	–	0.191	0.002	0.002	0.594	0.290	0.325	0.007	0.060
DR		–	< .001	0.688	0.018	< .001	0.007	1.000	< .001
DS			–	< .001	0.096	0.883	1.000	< .001	0.334
EV				–	< .001	< .001	< .001	1.000	< .001
GR					–	1.000	1.000	0.003	1.000
GS						–	1.000	< .001	1.000
KS							–	< .001	1.000
PR								–	< .001
SC									–

Table 11. Pairwise p-values for selection error under more faces condition. Values are shown only in the upper triangle for clarity.

© 2026 IEEE. This is the author's version of the article accepted for publication in IEEE TVCG.
The final version is available at: https://doi.org/10.1109/TVCG.2026.3680697

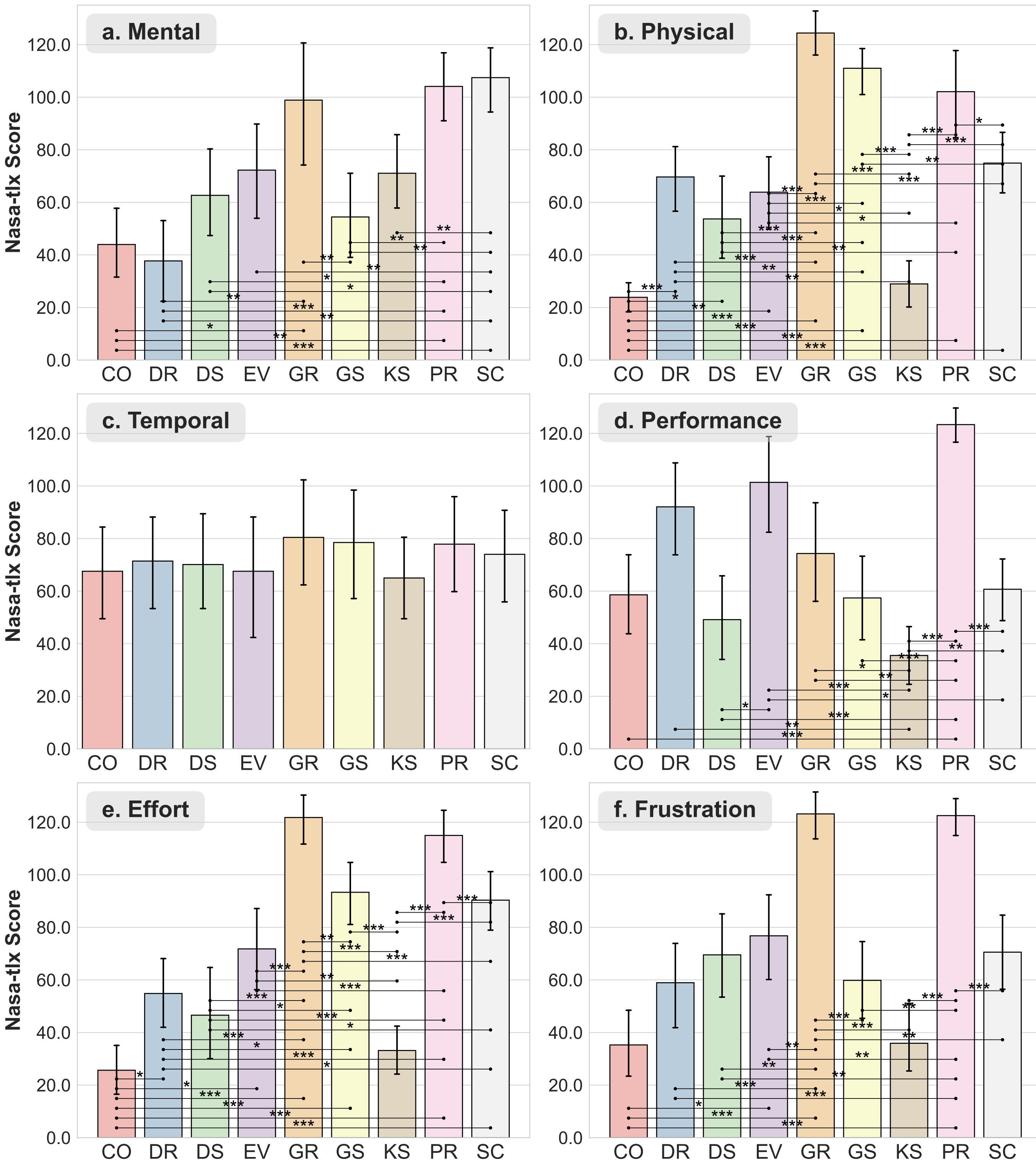

Fig. 8. Subjective measurement plots: (a–h) present the NASA-TLX scores across different dimensions. Lower scores indicate lower perceived workload for the technique, and thus better performance. Error bars indicate the standard error. Significant effects are marked ($* = p < .05$, $** = p < .01$, and $*** = p < .001$).