Designing Contextual Toolbars for Mixed Reality

Amir Azimyzadeh
amir.az@unb.ca
University of New Brunswick
Fredericton, New Brunswick, Canada

Colin Brett
colin.brett@unb.ca
University of New Brunswick
Fredericton, New Brunswick, Canada

Fahimeh Tolouee
fahimeh.tolouee@unb.ca
University of New Brunswick
Fredericton, New Brunswick, Canada

Cole K. Campbell
cole.campbell@unb.ca
University of New Brunswick
Fredericton, New Brunswick, Canada

Scott Bateman
scottb@unb.ca
University of New Brunswick
Fredericton, New Brunswick, Canada

Abstract

Contextual toolbars are toolbars that are spatially connected near the object they control, providing immediate and easy access to relevant functionality. In 3D mixed reality, however, issues related to their placement, including occlusion and reachability introduce new challenges that remain underexplored and can lead to usability problems. To address this gap, we conduct a series of studies, finding that the two leading approaches, object- and user-referenced toolbars, only work well in specific, opposing task scenarios. To address this, we introduce a novel hybrid toolbar that combines the features of the object- and user-referenced approaches, allowing the toolbar to be dynamically anchored to objects or detached from them to follow the user. Our hybrid approach balances efficiency and spatial context, remaining performant across different task scenarios. Our results contribute a new flexible hybrid interaction technique and new design guidelines for toolbars in mixed reality.

CCS Concepts

• **Human-centered computing → Interaction techniques**; **Mixed / augmented reality**.

Keywords

mixed reality, augmented reality, toolbar, contextual toolbar, interface design, human-computer interaction, spatial user interfaces

ACM Reference Format:
Amir Azimyzadeh, Colin Brett, Fahimeh Tolouee, Cole K. Campbell, and Scott Bateman. 2026. Designing Contextual Toolbars for Mixed Reality. In *Proceedings of Graphics Interface 2026*. ACM, New York, NY, USA, 11 pages.

1 Introduction

Originally designed for 2D interfaces, toolbars are a common interface interaction technique. Toolbars are most often organized as a list of icons that invoke commands when selected. *Contextual toolbars* are toolbars that are associated with a target object, most often being spatially positioned close by. For example, highlighting text in a document in MS Word reveals a contextual toolbar providing access to font related commands that can be applied to the selected text. Toolbars are also common in 3D mixed reality (MR) interfaces, where they are used to support a wide range of interactions, from object manipulation to system control [1, 20]. Most often, toolbars in MR are direct analogs of their 2D counterparts, where a flat 2D plane is placed in the 3D space [19]. However, in the move to 3D space of mixed reality, their design and implementation becomes inherently more complex. Unlike desktop environments, with static 2D screens, a user's ability to see and reach commands depends on where the menu is positioned relative to themselves [10, 19].

Common placement strategies for contextual toolbars include *object-referenced* (in which a toolbar is placed relative to the target it's associated with) and head- or body-referenced menus (in which the toolbar is placed relative to the user; we refer to these approaches collectively as *user-referenced* [11, 19]). These placement alternatives create a fundamental design trade-off. Object-referenced interfaces preserve a strong association between commands and their targeted point of interest, such as a 3D graphical object, due to their spatial proximity [19]. User-referenced interfaces, in contrast, travel with the user [11], allowing easier access to commands while the user looks and moves around the environment [19]. Although these trade-offs are widely recognized, it is unclear how they affect performance and experience in realistic MR workflows that extend beyond isolated command selections.

This gap is important because many MR use cases are not single interactions, but complex workflows. Some require users to alternate between multiple objects and repeatedly return to earlier targets, while others involve completing a sequence of actions on one object before moving to the next. These different task structures place different demands on attention, movement, and memory, suggesting that no single toolbar placement strategy is likely to perform best in all cases.

To investigate this, we first ran two preliminary studies to establish the best performing design strategies for both object- and user-referenced toolbars, in which participants identified target objects and used a toolbar to perform indirect commands on them. However, recognizing the trade-offs between the object-referenced and user-referenced approaches, we theorized about the scenarios where each would perform best: *action-focused* scenarios, which require revisiting multiple objects, and *sequence-focused* scenarios, which involve completing a sequence of actions on one object at a time. Object-referenced toolbars should better support the action-focused scenarios because they provide persistent menus attached

GI '26, Waterloo, ON, Canada

to each object, which can be quickly revisited to access information and invoke commands. In contrast, user-referenced toolbars should enable faster and less physically demanding interaction in sequence-focused scenarios, as they allow users to keep the toolbar in view while shifting attention between the object and an external information source (e.g., instructions).

Motivated by this design tension, we introduce a *hybrid toolbar* that allows users to dynamically switch between object- and user-referenced modes. The hybrid toolbar lets users adapt the interface to the demands of the current task. In our main study, participants used a toolbar to apply commands (e.g., resize, rotate, translate) to identifiable 3D cubes while consulting an instruction plate, under either a sequence-focused workflow or an interleaved workflow. We show that the theorized design tension between object- and user-referenced approaches actually exists, and that each approach struggles where the other excels. Additionally, we show that the flexibility of the hybrid toolbar helps balance the demands and maintains consistent performance across different task characteristics. We also find that the hybrid toolbar is strongly preferred by participants overall, despite requiring additional mode-switching interactions.

This paper makes three main contributions: (1) empirical evidence showing how toolbar placement interacts with task structure in MR, (2) a novel and simple hybrid toolbar technique that supports on-demand switching between user- and object-referenced modes, and (3) design implications for contextual menus that emphasize adaptability to workflow demands.

2 Related Work

2.1 Menus in Mixed Reality

A menu is a visually structured interface mechanism that presents a bounded set of commands for user selection, typically appearing on demand and disappearing after use [2]. Toolbars are often thought of as being persistent, but contextual menus and toolbars often appear only when an action is performed on a target object that indicates the need for the commands in the menu (e.g., right-clicking on a graphical object). Menu designs in MR have more variation than what is commonly seen in 2D desktop menus. These variations can be characterized by their placement, shape, and the interactions they require [8, 22, 26]. LaViola et al. categorize MR menu designs based on how user movement is interpreted to determine command selection [19]. 3D widgets leverage all three spatial dimensions, embedding selectable elements within a volumetric structure where depth, height, and width all contribute to interaction. For example, a 3×3×3 command cube [14] or scaling handles attached directly to a 3D object [7] relies on full spatial positioning to finalize the selection or manipulation. 2D menus adapted to 3D, in contrast, remain largely unchanged. 2D adapted menus preserve a planar layout in which commands (often represented by icons) are arranged on a flat surface and the selection action operates in a fundamentally two-dimensional way even when placed in 3D space [3]. Finally, 1-DOF (one-degree-of-freedom) menus restrict selection to a single spatial dimension, regardless of the user's overall movement; for instance, rotating the wrist or sliding the hand linearly may cycle through options arranged along a single axis [19]. Among these categories, adapted 2D menus are the most common choice in MR due to their familiarity to users and their ability to handle large, complex sets of functions [1, 20]. However, MR menu design presents unique challenges because, unlike menus in 2D environments, the possibilities for their placement and how they are interacted with is much larger [19, 21]. In this work, we focus specifically on toolbars as a representative form of such planar menu structures.

2.1.1 Positioning MR Menus.

The physical location of a menu in 3D space is a critical consideration for their design. The two primary design alternatives (i.e., object-referenced and user-referenced) create fundamental trade-offs in how they support users. While object-referenced interfaces preserve a strong association between commands and their point of reference, user-referenced menus offer easier access as they move with the user, allowing them to be both visible and reachable regardless of the user's viewpoint [19].

The accessibility to object-referenced menus can be inconsistent. Scaling virtual objects can occlude a menu placed next to an object, and unless handled carefully, interactions can become difficult or impossible if the object is large or distant. Consider a menu that is attached to the bottom edge of a virtual cube being resized; as the cube grows, the menu may become unreachable or even hidden from the user's view [19]. Some approaches explore dynamic or adaptive positioning to mitigate reachability issues [5, 28]. However, even with such techniques, maintaining consistent accessibility while preserving spatial attachment to the object remains an important design challenge.

User-referenced interfaces travel with the user [11], allowing easier access to commands while the user looks and moves around the environment [19]. For this reason, user-referenced menus provide more predictable spatial positioning and reachability relative to the user [29]. The advantages of user-referenced menus also come with two trade-offs. First, they often fail to indicate which object is currently being manipulated, which weakens the spatial relationship between the interface and the target object [19]. Second, a user-referenced menu would typically mean that only one menu is being used at a time, even if it is associated with a particular object. This, for example, would limit the ability to compare the options selected in a properties menu for two different objects [20, 29]. While both user-referenced and object-referenced menus have their advantages and disadvantages, it is unclear how these trade-offs affect user performance and experience across different task contexts in MR.

2.1.2 Making Connections between Menus and Objects.

One of the main challenges in MR is unstable visibility. Some of this is inherent, as virtual content can be distributed around the user; additionally, HMDs show only a limited portion of the scene at any time [13], so it is difficult to interact with objects that are partially visible or outside the user's view [30]. Prior work has addressed this primarily through out-of-view awareness techniques. Established off-screen visualizations, such as arrows and halos, have been adapted to MR to encode directional information at display boundaries [16]. Subsequent work has extended these approaches to fully 3D environments, using techniques such as 3D wedges, minimaps, and radar-style indicators to help users locate out-of-view targets [27, 31]. These findings suggest that while user-referenced toolbars lack inherent spatial context, visual cues like spherical wedges can help bridge the gap by guiding attention to the relevant target.

2.2 Tasks and Toolbars in Mixed Reality

The type of task that people perform might dictate which toolbar design provides the best support. Researchers have categorized two task types that can be performed with menus in MR: *geometric manipulation* and *abstract manipulation*. Geometric manipulation involves actions that directly modify a virtual object (e.g., scaling, rotating, or translating a 3D model) [4, 12]. With abstract manipulation, the interaction is often decoupled from a digital representation of a physical object (e.g., interacting with a website in MR) [17, 23, 24]. While toolbars in 3D environments can support both types, our work is not strictly limited to either category. In tasks of this kind, the placement of menus can be tricky because, as previously mentioned, object-referenced menus can become occluded or unreachable [19] and user-referenced menus can lead to confusion about which object is currently selected [29].

Although a wide range of toolbar placement strategies have been explored, many empirical evaluations focus on controlled ‘unit’ tasks such as simple item selection [2, 20, 29]. However, real-world work tasks and workflows often involve many such unit tasks. Further, they also involve “context switching,” where a shift from a primary task to a toolbar for indirect input can negatively affect speed and attention [9, 15].

It remains unclear how menu placement trade-offs are experienced across more complex interaction workflows, for example, in those where users must alternate between subtasks to perform object manipulation over the course of a larger task. This gap motivates our investigation of user-referenced and object-referenced toolbar designs across distinct task types to better understand how toolbar designs affect user performance.

3 Contextual Toolbar Design

In this section, we describe how basic task characteristics can be differently supported by the two main contextual toolbar design approaches. Contextual toolbar designs have been largely differentiated by their placement strategies. Optimal toolbar placement depends on variations in task characteristics. We distinguish two general ways that tasks can be characterized (as sequential tasks or interleaved tasks), and how they are supported by two different toolbar designs (user-referenced or object-referenced).

Sequential Tasks: Tasks involving executing a sequence of actions on a single object at a time, before moving to the next object. For example, a task might require editing a 3D model, or annotating a document in MR. In many applications, the actions to be performed on the work object itself are not done through direct manipulation alone, but often require additional commands through menus external to the object. In sequential tasks, user-referenced toolbar placement may be more effective; as the toolbar moves with the user, commands remain accessible while attention shifts between work objects, reducing the repeated reorientation required to interact with different menus.

Interleaved Tasks: Tasks that involve switching between multiple work objects. For example, a task might involve manipulating multiple objects, such as arranging virtual furniture in a room or editing multiple 3D models. In these scenarios, object-referenced toolbar placement is likely to be more effective; as the toolbar remains spatially attached to its associated object, it provides a persistent visual marker that helps users track which objects have been manipulated and which still require attention. This spatial consistency can reduce cognitive load and improve task efficiency by minimizing the need for users to repeatedly reorient themselves to locate the relevant interface for each object.

Hybrid Tasks: Real-world workflows often alternate between interleaved and sequential task structures, creating a situation where both object-referenced placement and user-referenced placement strategies are not overly effective. This observation informed our design of a novel *hybrid toolbar*, which combines elements of both object-referenced and user-referenced approaches by allowing users to switch between the two modes as needed. The hybrid toolbar is designed to provide the flexibility to adapt to different task contexts, enabling users to leverage the benefits of both placement strategies.

While we believe that our proposed hybrid toolbar should be effective for hybrid tasks, it is unclear whether they could provide meaningful benefits over the best-performing object-referenced and user-referenced designs.

4 Preliminary Studies

The main goal of our work is to investigate whether our hybrid toolbar design can provide meaningful benefits over existing approaches across different task contexts. However, before conducting a comparative evaluation of the hybrid toolbar, we first performed two preliminary studies to identify the best-performing object-referenced and user-referenced toolbar designs. These preliminary studies allowed us to refine our designs and ensure that we were comparing the most effective implementations of each approach in our final comparative evaluation with the new hybrid toolbar.

Below we describe the two preliminary studies, presenting them as tersely as possible for brevity, saving space to report more details on the main study. In all three studies (the two preliminary studies and the main study), our procedures were approved by the University of New Brunswick Research Ethics Board (REB #2024-021).

4.1 Visual Links in User-referenced toolbars

For the first preliminary study, we focused on the main challenge of user-referenced toolbars, which is maintaining spatial context between the toolbar and its associated object. User-referenced toolbars are placed relative to the user. However, this also means that toolbar is spatially detached from the object it controls, which can create ambiguity, particularly when multiple targets are present.

4.1.1 Conditions. To find the best approach for maintaining the spatial connection between the toolbar and its target object, we evaluated four techniques, described below (Figure 1):

Camera-Aligned Bounding Box: A bounding box is calculated relative to the headset’s viewport and is rendered to create a rectangular frame around the object. If the object moves out of the user’s field of view, the rectangle remains fixed to the nearest edge of the viewport to indicate the object’s relative direction.

Spherical Wedge: Adapted from the original Wedge technique [18], our version keeps the baseline of the wedge vertical relative to the user’s screen to provide a stable anchor for the toolbar. Unlike the original implementation, our version always points to the side of the viewport rather than upward or downward, which we found more natural, given how users typically rotate their heads.

Bézier Curves (Single and Double): While the Bounding Box and Wedge exist on a 2D plane or sphere close to the user, both Bézier techniques use 3D curves connected directly to the virtual object. These curves provide depth cues; the curvature indicates the distance to the target, and in the **Double Bézier** variation, the distance between the two lines represents the rotational distance.

4.1.2 Task. Participants ($N = 16$, 9 male, 7 female; ages 21–45) stood centrally among 50 distributed virtual objects. In each trial, the system selected a target object randomly. Participants used the assigned visual cue to locate the target and select its attributes on a user-referenced toolbar.

4.1.3 Findings. No significant differences were observed in the number of errors or amount of rotational movement required. However, the Spherical Wedge interface was significantly faster ($\chi^2(3) = 17.07, p < .001$) and most preferred. Consequently, the Spherical Wedge was chosen for the main study.

4.2 Object-referenced toolbar positioning

For the second study, we focused on object-referenced toolbars, and their biggest challenge, which is placing the toolbar relative to the associated object. Object-referenced toolbars rely on spatial proximity to their associated objects. However, when an object is large or positioned in a way that obstructs the toolbar, it can be difficult to view or access the toolbar.

In our implementation a toolbar appears near the associated object, after object selection, and can be dismissed by selecting the close button. We compared a baseline positioning strategy with a dynamic alternative that used a simple heuristic to try and minimize occlusion and increase the ease of access.

4.2.1 Conditions. Placement Conditions: The **MRTK Baseline**—the default behavior in Microsoft's Mixed Reality Toolkit—fixes the toolbar to the bottom edge of the object's bounding box. Our proposed **Height Heuristic** behavior dynamically snaps the toolbar to either the top or bottom edge based on the vertical distance between the user's head and the object, switching at approximately 80% of the user's height. This threshold was established through pilot studies to optimize ease of access to the bar regardless of its placement or the object's size (See Figure 1).

Scaling Conditions: In some MR applications, scaling a virtual object may additionally scale the size of a contextual toolbar. This makes sense, as really large objects may be viewed at a distance, but a contextual toolbar that does not scale up would be difficult to see and interact with at a distance. So, in addition to the placement strategy, we also evaluated the impact of dynamic scaling of the toolbar for both the MRTK Baseline and the Height Heuristic. In the **Scaling** condition, the toolbar size was adjusted linearly in proportion to the object's scale, while in the **No Scaling** condition, the toolbar maintained a constant size regardless of object scaling.

4.2.2 Task. : Four objects were randomly positioned at varying heights around the user. The experimental task involved selecting a series of commands from an instruction plate. Participants ($N = 20$, 12 male, 8 female; ages 19–41) completed trials under all four conditions (2 × 2; Placement × Scaling).

4.2.3 Results. : The **Height Heuristic** toolbar performed faster and required less user movement than **MRTK Baseline**, and that both conditions performed better overall without scaling; with faster completion times, reduced movement, and fewer errors. Participant feedback also indicated a general preference for the **Height Heuristic** toolbar. See the results in Table 1.

Table 1: Study 2 Results (Mean ± SD)

Measure	Scaling	Height Heur.	MRTK	$t(20)$	p
Time (s)	Non-scaling	21.6±5.6	23.8±6.8	−2.27	.034
	Scaling	23.8±6.6	21.7±5.8	3.88	.001
Movement (m)	Non-Scaling	6.63±1.58	7.66±1.91	−4.60	<.001
	Scaling	7.49±1.73	6.79±1.84	3.00	.007
Errors	Non-scaling	0.29±0.16	0.33±0.15	−0.86	.402
	Scaling	0.41±0.15	0.22±0.16	5.79	<.001

5 Main Study: Hybrid Toolbar Evaluation

In order to understand whether the hybrid toolbar design provides meaningful benefits over existing approaches, we conducted a comparative evaluation of the three toolbar designs (object-referenced, user-referenced, and hybrid) across both sequential and interleaved task contexts. Our study aimed to empirically assess how different toolbar placements affect user performance, workload, and overall experience in complex MR workflows that require both types of task structures. We compared the hybrid toolbar against the best-performing object-referenced and user-referenced implementations identified in the preliminary studies.

5.1 Design of the Hybrid Toolbar

Building on our preliminary findings, we developed a *hybrid toolbar* that synthesizes the strengths of both the user-referenced and object-referenced approaches. A single toolbar paradigm cannot optimally serve real-world tasks, as they often combine both sequential and interleaved structures.

The hybrid toolbar defaults to object-referenced behavior: when an object is selected for the first time, an object-referenced toolbar appears beside it and persists until explicitly closed by the user. Any object-referenced toolbar can be converted to user-referenced mode by pressing a pin icon located next to the close button in the toolbar's top-right corner. Once detached, the toolbar remains in the user's field of view while maintaining a visual connection to its target object via a spherical wedge. If another object is selected while the toolbar is in user-referenced mode, the toolbar redirects its visual connection to the newly selected object, and an object-referenced toolbar appears beside the previously targeted object. Pressing the pin icon again while in user-referenced mode re-anchors the toolbar to the currently targeted object. These interactions allow users to switch between object-referenced and user-referenced behavior as the task demands (see Figure 3).

5.2 Task Design

The evaluation was conducted in a physical room augmented with virtual content. Participants performed actions in two separate scenarios using a setup consisting of an *instruction plate* and a set of ten randomly generated *3D cubes*.

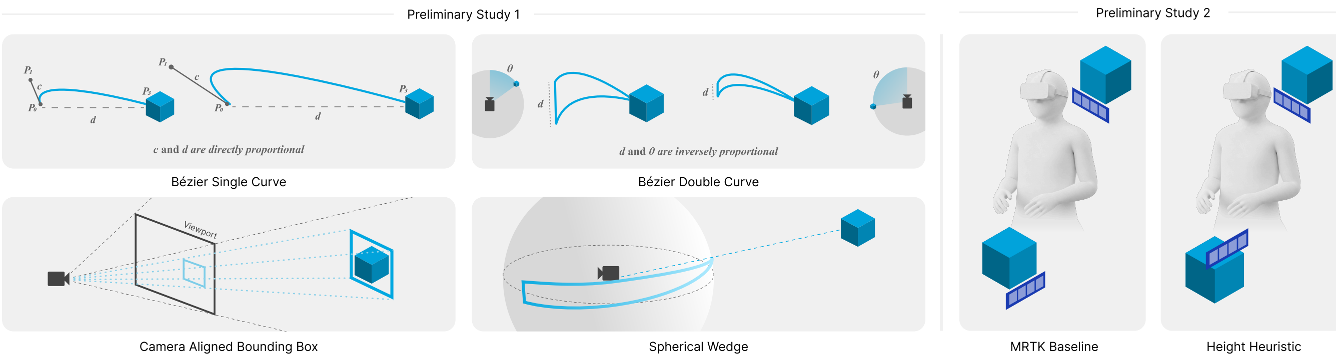

Figure 1: Conditions in the preliminary studies. Left: Study 1's four visual connection methods; d is user-to-target distance, and the Double Bézier adds rotational distance. Right: Study 2's placement strategies, showing how the Height Heuristic snaps the toolbar to the object's top edge to stay near the user, unlike the static MRTK Baseline.

Each cube displayed a unique identifier on its surface: a 5×5 grid of circles, where the number of filled circles corresponded to the cube's ID (see Figure 2). The instruction plate listed five target cube IDs along with a prescribed sequence of actions to perform on each indirectly using the toolbar (e.g., resizing, rotating, and translating in a specified direction). Target cube IDs, their corresponding actions, and the surface textures were all randomly generated for each block. Cube positions were randomized at the start of each block and remained fixed until the scenario was completed. We designed two distinct task scenarios to evaluate the strengths of each toolbar placement approach.

Our task does not fall cleanly into either the geometric or abstract category discussed earlier. The toolbar issued commands directed at a specific virtual cube, which keeps the spatial coupling between command and target that defines geometric manipulation. What matters for our investigation is the spatial coupling between the toolbar and its target, which is present regardless of the manipulation outcome.

Sequence-Focused Scenario (SFS): This scenario operationalized sequential task structure, in which a user completes a series of actions on a single object before moving on to the next. Participants worked through five target cubes, completing a prescribed sequence of three uninterrupted actions on each cube before advancing to the next cube. Each action required consulting the instruction plate, mirroring real sequential workflows where interaction with an object is mediated by commands or references external to it. In the user-referenced condition, the toolbar traveled with the user, allowing simultaneous access to both the instruction plate and the toolbar interface (see Figure 2). In the object-referenced condition, the toolbar remained anchored to the cube and moved out of view when participants shifted their gaze toward the instructions, requiring repeated visual reorientation to resume interaction. This scenario was intended to highlight the efficiency benefits of user-referenced placement when users must frequently shift attention between the work object and an external information source.

Action-Focused Scenario (AFS): This scenario operationalized interleaved task structure, in which users switch between multiple work objects and must track the state of each across time. We instantiated this structure through a two-stage operation on five target cubes: each cube required one action per stage, and Stage 2 was locked until Stage 1 had been completed for all targets. Within each stage, participants were free to address the cubes in whatever order they chose. This constraint forced participants to disengage from each object after Stage 1 and later return to it, producing the revisitation and state-tracking demands characteristic of interleaved work. The persistent toolbar provided by the object-referenced condition could serve as a visual marker indicating which cubes had already been handled, reducing the need to re-identify objects on return visits. The user-referenced condition, lacking any object-level cue, required participants to re-identify each cube by recounting filled circles, increasing the cognitive cost of interleaved task scenario.

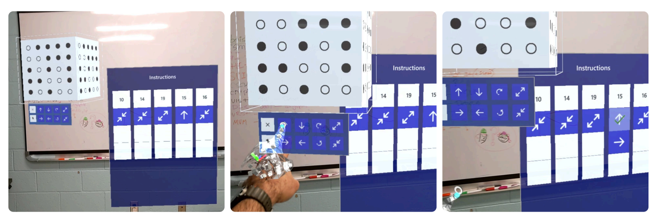

Figure 2: A single action performed on a target cube. Black circles represent a cube's ID. The next required action is shown on the instruction plate in the Sequence-Focused Scenario.

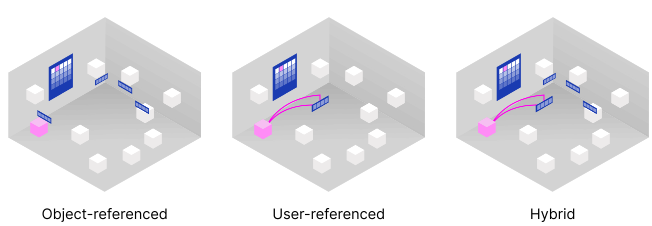

Figure 3: Conditions in the Main Study: object-referenced, user-referenced, and hybrid toolbars. In hybrid mode, one toolbar follows the user in user-referenced mode while others remain anchored to their respective objects.

The hybrid toolbar was designed to resolve the trade-off between these two paradigms. It leverages object-referenced spatial proximity to serve as a visual marker for previously visited objects, while providing the mobility needed to avoid gaze shifts during instruction-mediated tasks (see Figure 3). However, it was unclear whether this flexibility would translate into improved performance across both task scenarios, or whether the added complexity of managing the toolbar's mode would introduce new challenges.

5.3 Procedure

The study followed a within-subjects design with two factors: *toolbar condition* (object-referenced, user-referenced, hybrid) and *task scenario* (SFS, AFS). Each participant completed all six combinations of condition and scenario, resulting in six experimental blocks. The six blocks were fully counterbalanced across participants using a balanced Latin square with six orderings.

Prior to the experimental blocks, participants completed a tutorial session lasting approximately 5–10 minutes. During the tutorial, participants practiced locating target cubes by counting filled circles and executing actions using the toolbar. Efficiency strategies were introduced, such as counting unfilled circles when most were filled to speed identification, and using the toolbar to mark completed objects. Participants who felt insufficiently prepared were allowed to repeat the tutorial before proceeding to the main trials. Tutorial data were excluded from the analysis.

No time limits were imposed on any block. After completing each block, participants filled out a NASA-TLX workload questionnaire along with items assessing their preferences and experience with the toolbar in that scenario. A post-experiment questionnaire was administered at the end of the session to elicit overall comparisons across the three toolbar conditions. No breaks were scheduled between blocks, though participants were encouraged to rest if needed. Including tutorials, experimental blocks, and questionnaires, sessions lasted approximately 1.5–2 hours.

5.4 Measures

We recorded completion time, error count, and total physical movement (distance traveled) for each block. Errors were defined as selecting an incorrect action from the toolbar or performing an action on a non-target cube (i.e., one not listed on the instruction plate). We additionally logged all discrete user interactions, including: object selections, toolbar button presses, and toolbar dismissals.

5.5 Apparatus

The study was conducted using a Microsoft HoloLens 2 headset running an application developed in Unity 2022.3.16f1 with MRTK 2.8.3.

5.6 Participants

The hybrid toolbar evaluation included 18 participants (15 male, 3 female; aged 18–59, $M = 28.72$, $SD = 10.45$) recruited primarily from the university campus. Fourteen had prior MR experience, with 7 using it frequently (>1/year). Participants received a $20 gift card. The University of New Brunswick Research Ethics Board approved the study (REB #2024-021).

5.7 Statistical Analysis

We conducted our statistical analysis using R [25]. To determine the appropriate modeling approach, we first assessed the normality of the residuals from a baseline linear model for each dependent variable using the Shapiro-Wilk test. Variables that satisfied the normality assumption ($p \geq .05$) were modeled using Linear Mixed-Effects Models (LMMs) with a Gaussian distribution. For variables that violated this assumption ($p < .05$), we utilized Generalized Linear Mixed-Effects Models (GLMMs) to directly model the underlying distributions and preserve the original scale of the data.

Specifically, continuous variables (e.g., time and distance) were modeled using a Gamma distribution with a log-link function. Discrete count data (e.g., interactions) were modeled using a Poisson distribution. To account for potential overdispersion, we assessed all count models using the `performance` package and transitioned to a Negative Binomial distribution when necessary.

In all models, *Condition* and *Task*, along with their interaction, were entered as fixed effects, while *Participant* was included as a random intercept to account for our repeated-measures design. We ensured model convergence across all analyses; if default optimizers in the `lme4` package failed, models were refitted using the `glmmTMB` package. Significance of the fixed effects and their interactions was evaluated using Type III Analysis of Variance (Wald χ^2 tests for GLMMs and F-tests with Satterthwaite's approximations for LMMs). When a significant main or interaction effect was observed, we conducted post-hoc pairwise comparisons using estimated marginal means (via the `emmeans` package) to assess the magnitude and direction of the effects, adjusting all p-values with the Holm-Bonferroni correction. For pairwise comparisons from generalized linear mixed models, we report exponentiated contrast ratios on the response scale as effect sizes; for linear mixed models, we report Cohen's d.

6 Results

6.1 Performance Results

For each measure we report the omnibus effects of condition, scenario, and the condition × scenario interaction, followed by the patterns observed within the AFS and SFS. Throughout this section, H, U, and O denote the hybrid, user-referenced, and object-referenced toolbar conditions. "Within-subset" refers to actions performed on objects listed on the instruction plate (intended task targets); "outside-subset" refers to actions on the remaining, non-target objects. All pairwise contrast ratios and significance markers are reported in Table 2; in the prose below we describe the patterns and refer the reader to the table for the exact values.

6.1.1 Completion Time. Completion time showed no main effect of condition ($\chi^2(2) = 1.71, p = .425$), but a significant effect of scenario ($\chi^2(1) = 7.42, p = .006$) and a significant condition × scenario interaction ($\chi^2(2) = 22.38, p < .001$). In the AFS no conditions differed. In the SFS the user-referenced toolbar was significantly faster than both other conditions, while the hybrid and object-referenced did not differ. See Figure 4 (left) and Table 2.

6.1.2 Distance Traveled. Distance traveled showed significant main effects of condition ($\chi^2(2) = 34.32, p < .001$) and scenario ($\chi^2(1) = 14.90, p < .001$), and a significant condition × scenario interaction

Table 2: Pairwise comparisons across all performance measures and scenarios. Cells report condition ratios where significant; "n.s." denotes non-significant. Outside-subset SFS selections report Cohen's d instead of ratios. Significance: * $p < .05$, ** $p < .01$, *** $p < .001$. H = hybrid, U = user-referenced, O = object-referenced.

	Action-Focused Scenario			Sequence-Focused Scenario		
Measure	H/U	H/O	U/O	H/U	H/O	U/O
Completion time	n.s.	n.s.	n.s.	1.43^{***}	n.s.	0.61^{***}
Distance traveled	1.77^{***}	n.s.	0.50^{***}	2.18^{***}	0.69^{**}	0.32^{***}
Total interactions	0.81^{**}	1.16^{*}	1.43^{***}	1.16^{*}	1.21^{**}	n.s.
Overall selections	0.65^{**}	n.s.	1.40^{**}	1.39^{*}	n.s.	0.75^{*}
Within-subset selections	0.63^{***}	n.s.	1.61^{***}	n.s.	n.s.	n.s.
Outside-subset selections	n.s.	n.s.	n.s.	$d = 1.52^{***}$	n.s.	$d = -1.44^{***}$
Dismissals	0.43^{***}	n.s.	2.96^{***}	0.19^{***}	n.s.	7.55^{***}
Total errors	0.72^{***}	1.29^{**}	1.79^{***}	1.35^{**}	1.48^{***}	n.s.
Within-subset errors	n.s.	n.s.	n.s.	n.s.	n.s.	n.s.

($\chi^2(2) = 6.92, p = .031$). The user-referenced toolbar required significantly less physical movement than other conditions in both scenarios. The hybrid and object-referenced did not differ in the AFS, but in the SFS the hybrid required significantly less movement than the object-referenced. See Figure 4 (right) and Table 2.

6.1.3 *Interaction Count.* The number of interactions (toolbar button presses and target cube selections) showed significant main effects of condition ($\chi^2(2) = 31.10, p < .001$) and scenario ($\chi^2(1) = 16.58, p < .001$), and a significant interaction ($\chi^2(2) = 19.91, p < .001$). In the AFS, the user-referenced toolbar generated significantly more interactions than both other conditions, and the hybrid produced more than the object-referenced — largely due to additional pin toggle use. In the SFS, the hybrid produced significantly more interactions than both other conditions, while user-referenced and object-referenced did not differ. When pin toggle interactions were excluded, no significant differences remained between any conditions in the SFS (all $p > .05$), confirming that the hybrid's elevated count was attributable to incidental pin toggle use rather than a fundamental increase in interaction overhead.

6.1.4 *Object Selections.* Because participants sometimes selected non-target objects to use the toolbar as a spatial reminder, we partitioned selections into three views: overall, within-subset (intended targets), and outside-subset (non-targets).

Overall selections. Significant main effects of condition ($\chi^2(2) = 16.12, p < .001$) and scenario ($\chi^2(1) = 5.30, p = .021$), and a significant interaction ($\chi^2(2) = 23.28, p < .001$). In the AFS, the user-referenced toolbar produced significantly more selections than both other conditions, with no difference between hybrid and object-referenced. In the SFS the pattern reversed: the user-referenced produced significantly fewer selections than both other conditions, and again hybrid and object-referenced did not differ.

Within-subset selections. A significant main effect of condition ($\chi^2(2) = 21.11, p < .001$) and a significant interaction ($\chi^2(2) = 9.27, p = .010$); no main effect of scenario ($\chi^2(1) = 0.12, p = .731$). In the AFS, the user-referenced toolbar produced significantly more target selections than both other conditions. This is expected, as the hybrid and object-referenced toolbars typically remained open while only a single instance of the user-referenced toolbar could be available at a given time. In the SFS no significant differences emerged, likely because participants who located a target completed all required actions on it before moving on.

Outside-subset selections. Significant main effects of condition ($F(2, 85) = 8.56, p < .001$) and scenario ($F(1, 85) = 21.98, p < .001$), and a significant interaction ($F(2, 85) = 5.64, p = .005$). In the AFS no conditions differed. In the SFS, the hybrid and object-referenced toolbars produced significantly more outside-subset selections than the user-referenced, with no difference between hybrid and object-referenced. This indicates that participants used the hybrid and object-referenced toolbars as spatial reminder tools during the SFS, sometimes selecting non-target objects after inspecting them to mark them as already visited and avoid redundant counting.

6.1.5 *Dismissals.* Dismissals showed significant main effects of condition ($\chi^2(2) = 65.82, p < .001$) and scenario ($\chi^2(1) = 28.37, p < .001$), and a significant interaction ($\chi^2(2) = 11.69, p = .003$). In both scenarios, dismissals were significantly higher for the user-referenced toolbar than for the other two conditions, which did not differ from each other. Although users could open a new user-referenced toolbar by selecting another object (which would refocus the menu), in most cases they preferred to dismiss the current toolbar before selecting the next target.

6.1.6 *Errors.* We analyzed errors at two levels: overall errors (incorrect actions on any object) and within-subset errors (incorrect actions on intended target objects only).

Overall errors. A significant main effect of condition ($\chi^2(2) = 49.69, p < .001$) and a significant interaction ($\chi^2(2) = 31.28, p < .001$); no significant main effect of scenario ($\chi^2(1) = 3.10, p = .078$). In the AFS, error counts were significantly higher for the user-referenced toolbar than for both other conditions, and the hybrid also produced more errors than the object-referenced. In the SFS the pattern reversed: the hybrid produced significantly more errors than both other conditions, while user-referenced and object-referenced did not differ. This elevated count is consistent with the incidental pin toggle behavior described above; accidental pin toggles followed by unintended actions on non-target objects likely contributed to this pattern.

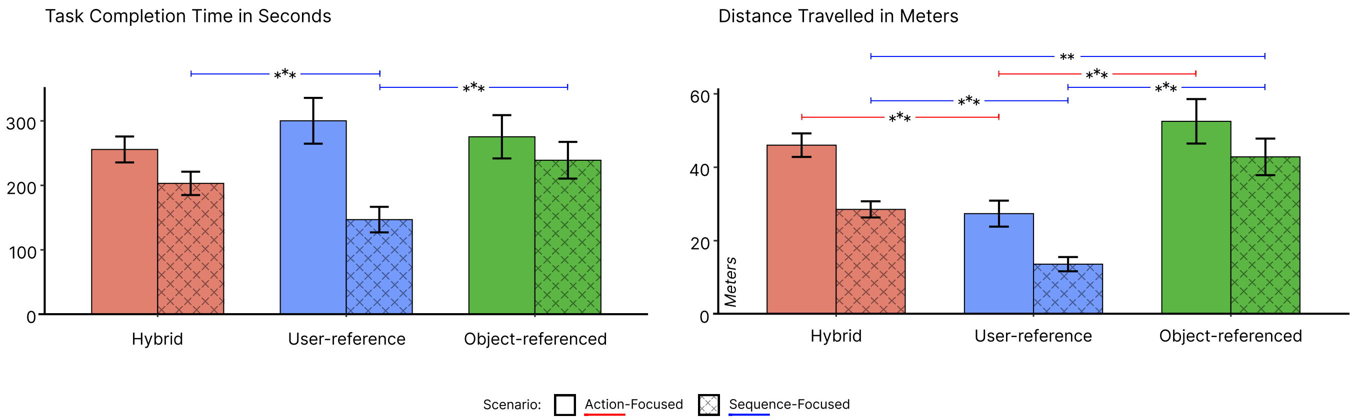

Figure 4: Task performance measures: (Left) Mean completion time and (Right) Mean distance traveled across the three toolbar conditions in Action-Focused and Sequence-Focused scenarios. Bars show Mean ± SEM.

Within-subset errors. No significant effects emerged for condition, scenario, or their interaction (all $p > .08$). When restricted to intended task targets, the three toolbar designs produced comparable accuracy in both scenarios.

6.2 User Experience Results and Observations

6.2.1 Participant Preference Rankings. After completing all tasks, participants ranked each toolbar across several subjective dimensions. The hybrid received the highest ratings on nearly every dimension, scoring significantly higher than both the user- and object-referenced toolbars for *Accuracy* ($p = 0.0166$ and $p = 0.0279$), *Overall Experience* ($p < 0.001$ and $p = 0.0193$), *Satisfaction* ($p = 0.0098$ and $p = 0.0419$), and perceived *Speed* ($p = 0.0010$ and $p = 0.0059$). The only exception was *Design Preference*, where the hybrid was rated lower than the user-referenced ($p < 0.001$) but higher than the object-referenced ($p = 0.0423$). See Figure 5.

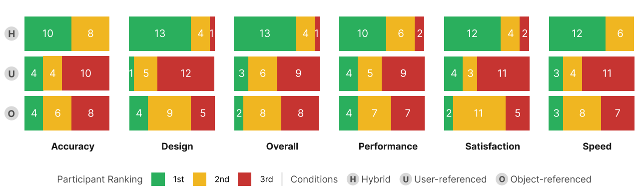

Figure 5: Main Study Condition Rankings

6.2.2 Likert Scale Responses. After each task-condition pair, participants answered Likert items about the interface. **AFS:** The object-referenced toolbar was rated as significantly less disruptive ($p = 0.0008$) and less distracting ($p = 0.0012$) than the user-referenced. Participants also rated it as more intuitive ($p = 0.0022$), and reported better understanding ($p = 0.0163$) and performance ($p = 0.0213$). The hybrid sat between the two: more disruptive than the object-referenced ($p = 0.0215$), but less distracting than the user-referenced. **SFS:** No significant differences emerged across conditions, suggesting that when participants worked continuously on a single object before moving on, all three toolbar designs were perceived as similarly usable.

6.2.3 NASA-TLX Workload Assessment. **AFS:** A single significant difference emerged: participants reported lower *Effort* ($p = 0.049$) with the object-referenced toolbar compared to the user-referenced. No other workload dimensions differed. **SFS:** Workload differences were more pronounced. Participants reported lower *Physical Demand* ($p = 0.002$) and *Effort* ($p = 0.005$) with the user-referenced toolbar than with both the object-referenced and hybrid. Overall workload was also lower for the hybrid than for the object-referenced ($p = 0.038$). See Figure 6.

6.2.4 Observational Findings and Participant Feedback. Several behavioral patterns emerged during the study that help contextualize the quantitative results.

Spatial organization strategies. Participants frequently used the object-referenced and hybrid toolbars as spatial markers. P8 noted that they used toolbars attached to objects to narrow their search space, and similar strategies appeared with the hybrid. This suggests that stationary toolbars acted as cognitive aids for task organization, particularly in the counting task where multiple objects had to be tracked.

View occlusion concerns. The user-referenced toolbar drew the most negative reactions. P2 remarked during the experiment, "I don't like this one, it's in my way," a sentiment echoed by several participants who described it as intrusive rather than helpful.

Hybrid mode use and adoption. Although the hybrid received the highest subjective ratings, observation revealed two distinct adoption patterns. Some participants treated it primarily as an object-referenced toolbar, only detaching it when occlusion or repetition became frustrating; others detached it proactively at the start of repetitive sub-tasks and re-anchored it when switching back to spatial work. Several participants explicitly cited the ability to choose between modes as the reason they preferred it overall, even when objective performance was comparable to one of the baselines. Initial confusion about mode switching was common but typically resolved within a few interactions.

6.3 Summary of Results

The main study yielded four key findings:

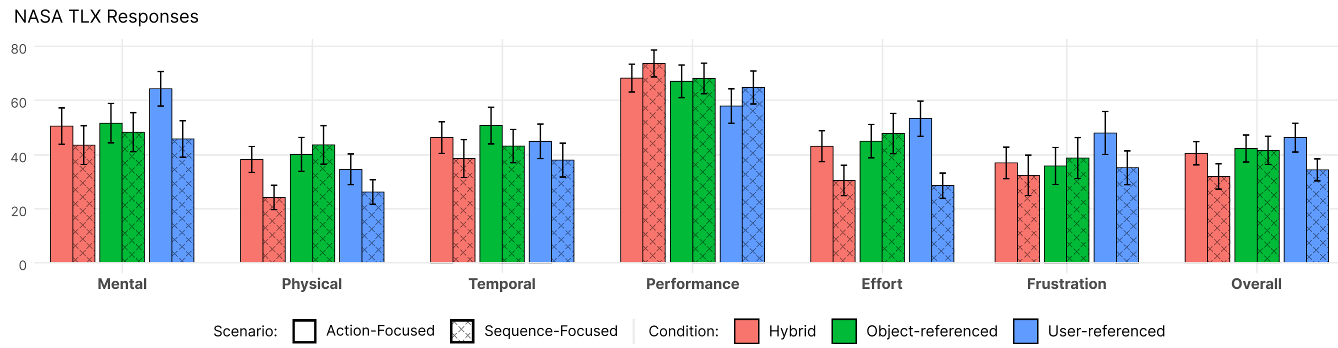

Figure 6: Mean (±*SEM*) NASA-TLX Taskloading Results for each Toolbar Condition and Task Scenario

(1) No toolbar design dominated across both scenarios; the user-referenced and object-referenced toolbars underperformed outside their optimal context, while the hybrid performed consistently across both but did not significantly outperform the best design in either scenario.
(2) Higher interaction counts and errors with the hybrid were attributable to incidental pin toggle use; within-subset accuracy was comparable across all three designs and scenarios.
(3) The hybrid received the highest rankings for *Accuracy*, *Overall Experience*, *Satisfaction*, and perceived *Speed*, rated significantly higher than both other designs on all four.
(4) Participants used the object-referenced and hybrid toolbars as spatial markers to track visited objects; the user-referenced toolbar was perceived as occluding the workspace.

7 Discussion

7.1 Which toolbar design is best?

Our findings showed that no single toolbar design dominated universally; each traditional design excelled in its theoretically predicted context while underperforming in the other. The hybrid toolbar consistently provided competitive performance across both contexts, avoiding the sharp degradation seen when single-design toolbars were applied outside their optimal scenario.

The results revealed the nuanced tradeoff between toolbar designs. In the action-focused scenario (AFS), completion time did not differ across conditions, but the user-referenced toolbar produced significantly more interactions, dismissals, and overall errors than both the object-referenced and hybrid toolbars. The object-referenced toolbar resulted in the fewest errors and interactions in this scenario. Physical movement, however, was significantly lower for the user-referenced toolbar, as it did not require users to travel to object locations. Notably, all three designs produced comparable error rates on intended targets; the user-referenced toolbar's elevated count was driven by errors on non-target objects.

In the sequence-focused scenario (SFS), these patterns largely reversed. The user-referenced toolbar was significantly faster and required the least physical movement, while the object-referenced toolbar demanded the most. The hybrid toolbar fell between the two on movement but did not differ from the object-referenced toolbar on completion time. Outside-subset selection patterns confirmed that participants used both the object-referenced and hybrid toolbars as spatial memory aids, selecting non-target objects to mark them as visited.

Subjective measures reinforced the scenario-dependent pattern. In the AFS tasks, participants rated the object-referenced toolbar as less disruptive, less distracting, and more intuitive, and NASA-TLX scores indicated lower effort. In the SFS tasks, workload ratings favored the user-referenced toolbar, with significantly lower physical demand and effort. Across both scenarios, the hybrid toolbar received the highest overall preference ratings for accuracy, satisfaction, speed, and overall experience, despite requiring additional mode-switching interactions. The one exception was design preference, where the user-referenced toolbar was rated highest.

Taken together, the results support the hybrid toolbar as a robust design that avoids the worst-case performance of either single-design approach. It matches the object-referenced toolbar in the interleaved context and narrows the gap with the user-referenced toolbar in the sequential context, without introducing meaningful accuracy costs on intended task objects in either scenario. The strong subjective preference for the hybrid toolbar, despite comparable objective performance, likely stems from two factors. First, it let participants avoid the worst case of either alternative — the clutter of a persistent user-referenced toolbar during interleaved work, and the repeated reorientation of a fixed object-referenced toolbar during sequential work. This seems to be the negativity bias at work, where worst-case experiences weigh more heavily in subjective ratings than averages. Second, participants valued the option to switch even when they rarely exercised it, suggesting that user agency and the feeling of control matters independently of how often a control is exercised.

7.2 How does toolbar usage inform design?

Participants used the object-referenced toolbar not only as an interaction surface but also as a spatial marker. Several explicitly described using open toolbars to narrow their search space. Because each toolbar remained anchored to its associated object, participants could offload the need to remember which objects had already been visited onto the environment itself. This effectively turned the toolbar into an external memory aid, reducing the need to re-identify targets by recounting filled circles. In contrast, the user-referenced toolbar existed as a single detached instance and provided no comparable spatial cues [28], contributing to the higher dismiss and selection counts observed in that condition. This points to a broader design implication: in multi-object workflows, persistent object-anchored interfaces can serve dual roles as both command interfaces and representations of task state.

The user-referenced toolbar, however, offered speed advantages in SFS task, but at a perceptual cost. Likert-scale ratings indicated that it was more disruptive and distracting in AFS task. Unlike desktop environments, where menus can be minimized or relegated to a taskbar, a user-referenced toolbar in MR occupies persistent space within the user's field of view [19]. When attention remained focused on a single object, this persistence was tolerable and even beneficial. When attention alternated between the instruction plate and the object — a form of context switching known to negatively affect speed and attention [9, 15] — it became obstructive. These results suggest that constant availability should not be treated as a default design assumption, and that toolbar visibility should instead adapt to the current phase of the task, consistent with prior proposals for dynamic and adaptive menu positioning [5].

Consistent with this, no significant differences emerged in Likert-scale responses across conditions in the SFS task. Participants perceived all three toolbars as similarly usable when the task did not require context switching between objects. This indicates that sequence-focused task structures are relatively forgiving of toolbar placement: when users operate on a single object within a linear workflow, the toolbar's spatial relationship to the environment becomes less consequential.

7.3 What about other toolbar designs?

Our investigation focused on two explicit spatial anchoring strategies. Namely, the user- and object-referenced toolbars are representative egocentric and allocentric techniques, along with a hybrid combining both. However, many other anchoring and design strategies exist that we did not examine, including object-agnostic world-referenced menus, fixed-view HUDs, hand-attached or gaze-driven menus, and various off-screen visualization techniques. Many of these alternatives can be situated along the same egocentric–allocentric dimension. A hand-attached menu, for example, is fundamentally egocentric and user-referenced. As such, our findings on when each anchoring strategy is effective may generalize to these alternative forms and inform future work on how such factors shape menu use. Future work could also explore implicit mode switching driven by gaze, dwell, or detected task, which may reduce the overhead created by our toggle interaction.

7.4 Limitations

There are several limitations to our study that may have effects on the generalizability of our results.

Participants - Our main study included 18 participants with a skewed gender distribution (15 male, 3 female). While our preliminary studies had more balanced samples (9M/7F and 12M/8F), and no gender-based differences were observed, this imbalance in the final study may limit the generalizability of the findings. Prior work has occasionally reported gender-related differences for tasks involving spatial memory and navigation [6], so our subjective preference results and observations around spatial-marker use should be interpreted with this caveat in mind. While our sample sizes are in line with comparable MR toolbar studies, they limit the strength of the claims we can make, particularly for null effects.

Hardware - All studies were conducted on a Microsoft HoloLens 2, an optical see-through headset with a limited field of view and hand-tracking-based input. Both of these hardware characteristics likely influenced our results. A wider field of view might reduce the occlusion issues observed with the user-referenced toolbar, and controller-based input might change the physical demand profile. Replication on other devices, including virtual reality headsets and future augmented reality hardware, would help establish the boundaries of our findings.

Task Breadth - A primary reason toolbars have persisted across systems is the breadth of tasks they support. In our study, all target objects were stationary, within arm's reach, and of manageable size. For example, in Preliminary Study 1, we discussed that scaling the toolbar might be beneficial to objects that are at a distance. These constraints leave several common usage scenarios unaddressed. When virtual objects are very large (e.g., allowing users to move through them) or located at a distance, the assumptions underlying both the Height Heuristic and near-interaction patterns may no longer hold. Similarly, when target objects are in motion, an object-referenced toolbar would need to move with them, potentially introducing challenges for targeting and readability. In such cases, a user-referenced approach may be more practical by default.

More broadly, our tasks focused on acting on a single object at a time, even in the AFS. An important extension is batch operations, where actions are applied to multiple selected objects simultaneously [19]. This raises open questions about where a shared toolbar should appear, how it should relate spatially to a set of targets, and what object-referenced anchoring means when interaction spans multiple objects.

Task Realism - The cube-identification mechanism introduced a working-memory demand that is not part of toolbar design itself. This load interacted with our findings: persistent toolbars in the object-referenced and hybrid conditions could be co-opted as spatial markers to offload it, likely amplifying their advantage in the AFS scenario. In workflows where targets are visually distinct or labelled, this benefit may be smaller.

8 Conclusion

This work investigates how toolbar design, specifically related to placement, affects user performance and experience in MR across two task scenarios: interleaved object manipulation and sequential action completion. Through two preliminary studies we identified design alternatives for the two leading toolbar placement approaches: object-referenced and user-referenced. Based on our findings, we developed a novel *hybrid toolbar* that combines the strengths of both approaches. In a comparative evaluation of all three toolbars, we showed that while no single placement strategy clearly dominates across task contexts, the hybrid toolbar maintained excellent performance across task scenarios and was the most preferred design across nearly all subjective dimensions. Additionally, we observed that the object-referenced toolbars supported spatial organization and acted as external memory aids during interleaved tasks, while user-referenced toolbars enabled faster and less effortful interaction during sequential tasks. However, in the case of the user-referenced and object-referenced toolbars, our results suggest that they can perform poorly under task scenarios that they do not match well. The hybrid toolbar provides a balanced solution that adapts to the demands of different task structures. Our results suggest that toolbar placement in mixed reality should not be treated as a fixed design decision; rather, it should adapt to the structure of the task it supports, and our hybrid approach offers a blueprint for designing adaptive toolbar systems.

9 Acknowledgement

We thank Ryan Groom with Kognitiv Spark for providing us with the idea of studying toolbars in MR. This work was partially supported by the Natural Sciences and Engineering Research Council of Canada (NSERC). We acknowledge the use of AI-assisted tools in the preparation of this paper. Gemini (Google) was used to assist in generating portions of illustrative figures for presentation clarity only; no AI-generated content was used in the scientific figures or results analysis. Additionally, Claude (Anthropic) was used for grammar, spelling, and writing assistance during manuscript preparation to improve language clarity and quality.

References

[1] Ian G. Angus and Henry A. Sowizral. 1995. Embedding the 2D interaction metaphor in a real 3D virtual environment. In *Stereoscopic Displays and Virtual Reality Systems II*, Scott S. Fisher, John O. Merritt, and Mark T. Bolas (Eds.), Vol. 2409. International Society for Optics and Photonics, SPIE, 282 – 293. doi:10.1117/12.205875

[2] Gilles Bailly, Eric Lecolinet, and Laurence Nigay. 2016. Visual menu techniques. 49, 4 (2016), 60:1–60:41. doi:10.1145/3002171

[3] D.A. Bowman and C.A. Wingrave. 2001. Design and evaluation of menu systems for immersive virtual environments. In *Proceedings IEEE Virtual Reality 2001*. 149–156. doi:10.1109/VR.2001.913781

[4] Nikolas Chaconas and Tobias Höllerer. 2018. An evaluation of bimanual gestures on the microsoft hololens. In *2018 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)* (2018-03). 1–8. doi:10.1109/VR.2018.8446320

[5] Yifei Cheng, Yukang Yan, Xin Yi, Yuanchun Shi, and David Lindlbauer. 2021. SemanticAdapt: optimization-based adaptation of mixed reality layouts leveraging virtual-physical semantic connections. In *Proceedings of the 34th Annual ACM Symposium on User Interface Software and Technology*. 282–297. doi:10.1145/3472749.3474750

[6] Emanuele Coluccia and Giorgia Louse. 2004. Gender differences in spatial orientation: A review. *Journal of Environmental Psychology* 24, 3 (2004), 329–340. doi:10.1016/j.jenvp.2004.08.006

[7] Brookshire D. Conner, Scott S. Snibbe, Kenneth P. Herndon, Daniel C. Robbins, Robert C. Zeleznik, and Andries Van Dam. 1992. Three-dimensional widgets. In *Proceedings of the 1992 symposium on Interactive 3D graphics* (New York, NY, USA, 1992-06-01) *(I3D '92)*. Association for Computing Machinery, 183–188. doi:10.1145/147156.147199

[8] Raimund Dachselt and Anett Hübner. 2007. Three-dimensional menus: A survey and taxonomy. *Computers & Graphics* 31, 1 (2007), 53–65. doi:10.1016/j.cag.2006.09.006

[9] Andre Doucette, Carl Gutwin, and Regan L. Mandryk. 2010. A comparison of techniques for in-place toolbars. (2010). hdl:1880/50685 doi:10.11575/PRISM/35539

[10] Barrett Ens, Juan David Hincapié-Ramos, and Pourang Irani. 2014. Ethereal planes: a design framework for 2D information space in 3D mixed reality environments. In *Proceedings of the 2nd ACM symposium on Spatial user interaction* (New York, NY, USA, 2014-10-04) *(SUI '14)*. Association for Computing Machinery, 2–12. doi:10.1145/2659766.2659769

[11] Steven Feiner, Blair MacIntyre, Marcus Haupt, and Eliot Solomon. 1993. Windows on the world: 2D windows for 3D augmented reality. In *Proceedings of the 6th annual ACM symposium on User interface software and technology* (New York, NY, USA, 1993-12-01) *(UIST '93)*. Association for Computing Machinery, 145–155. doi:10.1145/168642.168657

[12] Zihan Gao, Huiqiang Wang, Hongwu Lv, Moshu Wang, and Yifan Qi. 2022. Evaluating the effects of non-isomorphic rotation on 3d manipulation tasks in mixed reality simulation. 28, 2 (2022), 1261–1273. doi:10.1109/TVCG.2020.3010247

[13] Kristen Grinyer and Robert J. Teather. 2022. Effects of field of view on dynamic out-of-view target search in virtual reality. In *2022 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)* (2022-03). 139–148. doi:10.1109/VR51125.2022.00032

[14] Jérôme Grosjean and Sabine Coquillart. 2001. Command & control cube: a shortcut paradigm for virtual environments. In *Immersive Projection Technology and Virtual Environments 2001* (Vienna, 2001), Bernd Fröhlich, Joachim Deisinger, and Hans-Jörg Bullinger (Eds.). Springer, 1–12. doi:10.1007/978-3-7091-6221-7_1

[15] Jens Grubert, Tobias Langlotz, Stefanie Zollmann, and Holger Regenbrecht. 2017. Towards pervasive augmented reality: Context-awareness in augmented reality. 23, 6 (2017), 1706–1724. doi:10.1109/TVCG.2016.2543720

[16] Uwe Gruenefeld, Abdallah El Ali, Wilko Heuten, and Susanne Boll. 2017. Visualizing out-of-view objects in head-mounted augmented reality. In *Proceedings of the 19th International Conference on Human-Computer Interaction with Mobile Devices and Services* (New York, NY, USA, 2017-09-04) *(MobileHCI '17)*. Association for Computing Machinery, 1–7. doi:10.1145/3098279.3122124

[17] Aakar Gupta, Bo Rui Lin, Siyi Ji, Arjav Patel, and Daniel Vogel. 2020. Replicate and reuse: tangible interaction design for digitally-augmented physical media objects. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2020-04-23) *(CHI '20)*. Association for Computing Machinery, 1–12. doi:10.1145/3313831.3376139

[18] Sean Gustafson, Patrick Baudisch, Carl Gutwin, and Pourang Irani. 2008. Wedge: clutter-free visualization of off-screen locations. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (New York, NY, USA, 2008-04-06) *(CHI '08)*. Association for Computing Machinery, 787–796. doi:10.1145/1357054.1357179

[19] Joseph J. LaViola, Ernst Kruijff, Doug Bowman, Ryan P. McMahan, and Ivan P. Poupyrev. 2017. *3D User Interfaces: Theory and Practice* (2 ed.). Addison-Wesley, Redwood City, CA, USA. 624 pages.

[20] Irina Lediaeva and Joseph J. LaViola. 2020. Evaluation of body-referenced graphical menus in virtual environments. In *Proceedings of Graphics Interface 2020* (2020). Canadian Human-Computer Communications Society / Société canadienne du dialogue humain-machine. doi:10.20380/GI2020.31

[21] David Lindlbauer, Anna Maria Feit, and Otmar Hilliges. 2019. Context-aware online adaptation of mixed reality interfaces. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology (UIST '19)*. Association for Computing Machinery, 147–160. doi:10.1145/3332165.3347945

[22] Lu Lu, Pengshuai Duan, Xukun Shen, Shijin Zhang, Huiyan Feng, and Yong Flu. 2021. Gaze-pinch menu: performing multiple interactions concurrently in mixed reality. In *2021 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)* (2021-03). 536–537. doi:10.1109/VRW52623.2021.00150

[23] Katsutoshi Masai, Kai Kunze, Daisuke Sakamoto, Yuta Sugiura, and Maki Sugimoto. 2020. Face commands-user-defined facial gestures for smart glasses. In *2020 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)* (2020-11). 374–386. doi:10.1109/ISMAR50242.2020.00064

[24] Erwan Normand and Michael J. McGuffin. 2018. Enlarging a smartphone with ar to create a handheld vesad (virtually extended screen-aligned display). In *2018 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)* (2018-10). 123–133. doi:10.1109/ISMAR.2018.00043

[25] R Core Team. 2021. *R: A language and environment for statistical computing*. R Foundation for Statistical Computing, Vienna, Austria. https://www.R-project.org/

[26] A. Santos, T. Zarraonandia, P. Díaz, and I. Aedo. 2017. A comparative study of menus in virtual reality environments. In *Proceedings of the 2017 ACM International Conference on Interactive Surfaces and Spaces* (New York, NY, USA, 2017-10-17) *(ISS '17)*. Association for Computing Machinery, 294–299. doi:10.1145/3132272.3132277

[27] Richard Stoakley, Matthew J. Conway, and Randy Pausch. 1995. Virtual reality on a WIM: interactive worlds in miniature. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (USA, 1995-05-01) *(CHI '95)*. ACM Press/Addison-Wesley Publishing Co., 265–272. doi:10.1145/223904.223938

[28] Markus Tatzgern, Denis Kalkofen, Raphael Grasset, and Dieter Schmalstieg. 2014. Hedgehog labeling: View management techniques for external labels in 3D space. In *2014 IEEE Virtual Reality (VR)* (2014-03). 27–32. doi:10.1109/VR.2014.6802046

[29] Yanbin Wang, Yizhou Hu, and Yu Chen. 2021. An experimental investigation of menu selection for immersive virtual environments: fixed versus handheld menus. 25, 2 (2021), 409–419. doi:10.1007/s10055-020-00464-4

[30] Amelia C. Warden, Christopher D. Wickens, Daniel Rehberg, Benjamin A. Clegg, and Francisco R. Ortega. 2025. Clutter costs in head-mounted displays: A study examining trade-offs between overlay and adjacent presentation of information. *Cognitive Research: Principles and Implications* 10, 1 (Aug. 2025), 47. doi:10.1186/s41235-025-00650-5

[31] Difeng Yu, Hai-Ning Liang, Kaixuan Fan, Heng Zhang, Charles Fleming, and Konstantinos Papangelis. 2020. Design and evaluation of visualization techniques of off-screen and occluded targets in virtual reality environments. 26, 9 (2020), 2762–2774. doi:10.1109/TVCG.2019.2905580