LookUP: Command Search Using Dwell-free Eye Typing in Mixed Reality

Jinghui Hu[*] John J. Dudley[†] Per Ola Kristensson[‡]

University of Cambridge, United Kingdom

(a) (b)

Figure 1: (a) The neutral head position while wearing the HoloLens2. The search icon is placed at the **body-fixed top** position. (b) The user moves their head and eye gaze to align with the search icon for activation of the virtual keyboard in both UPDWELL and UPDWELLFREE.

ABSTRACT

We introduce LookUP, a novel general purpose command search system for mixed reality headsets, offering a hands-free experience through dwell-free eye typing. With LookUP, users can trigger the display of a virtual keyboard with a simple upward head motion. The keyboard then uses a statistical decoder to interpret users' intended text based on their eye movements. This approach diverges from traditional dwell-time methods, significantly enhancing typing speed and efficiency. Our research involved deploying LookUP on a HoloLens 2, and benchmarking it against a dwell-based command search baseline and the native HoloLens system menu. Our user study indicated that participants spent a significantly shorter time using LookUP with dwell-free eye typing in command search and entry, demonstrating LookUP's potential to be a complementary command input for mixed reality headsets.

Index Terms: Mixed Reality, Command search, gaze tracking.

1 INTRODUCTION

The advent of Mixed Reality (MR) headsets has introduced new challenges for command entry due to their limited input capabilities. Traditional methods such as menu-based command selections and mid-air gestures have been extensively studied [7,9,37,42], but they often require intricate interactions or the memorization of gestures. Additionally, command search and selection are versatile, applicable to various tasks such as operating systems [29] to text editing [28] and 3D sketch editing [47]. These tasks typically require a proactive shift from other activities.

This paper introduces LookUP, a novel hands-free command search system designed to enhance user experience in MR environments. LookUP allows users to activate a virtual keyboard with a simple upward head motion, enabling dwell-free eye typing for command search. When the user's head and eye gaze align with the search icon, the system triggers the virtual keyboard (Fig. 1). This design minimizes the need for extensive prior knowledge of the user interface, making it intuitive and efficient. The system uses a statistical decoder to interpret gaze sequences, significantly enhancing typing speed and reducing eye strain compared to traditional dwell-time methods.

In this paper, we explore the potential of a hands-free command search system designed to complement and work alongside conventional menu-based methods. Our goal is to support users in scenarios where their hands are occupied, such as when operating machinery in a manufacturing plant. Additionally, this system provides a subtle interaction method in public spaces, reducing the conspicuousness associated with mid-air gestures or speech input.

Eye-based menu activation [33] and dwell-free eye typing [23] offer an alternative for MR interactions that could be more efficient in specific contexts [17]. Leveraging this insight, the system LookUP allows users to quickly find and execute commands through dwell-free eye typing. This paper explores the gap in understanding how eye gaze can be used to assist or provide other means for users to achieve their specific goals in practical mixed reality tasks.

In summary, our work makes the following contributions:

- We present LookUP: a command search system for mixed reality that leverages the dwell-free eye typing paradigm.
- An empirical user study demonstrating that the LookUP system allows users to complete command search and selection tasks 30% faster than a conventional hand-based menu system.

2 RELATED WORK

2.1 Information Search

Searching for information has become a seamless part of everyday life, enhanced by the evolution of technology that facilitates easy access to vast data resources. The introduction of intuitive search interfaces, from internet search engines like Google and Bing to local file searches like the Microsoft Windows Start Menu and Apple macOS Spotlight, has transitioned us from using complex query languages to natural language inputs. This simplification allows

---

[*] e-mail: jh2265@cam.ac.uk
[†] e-mail:jjd50@cam.ac.uk
[‡] e-mail:pok21@cam.ac.uk

users to type, speak, or visually input queries using simple keywords or questions. Modern search technologies, including voice-activated assistants like Siri and Alexa, leverage sophisticated algorithms to fetch results based on relevance, recency, and user preferences, making digital information not only accessible but tailored to individual needs. This continuous refinement of search mechanisms has turned searching into an intuitive, integral tool for knowledge discovery, indispensable in both personal and professional contexts.

Prior research on information retrieval in virtual or mixed reality environments mainly focused on searching for 3D objects [11,13,49], images [51], videos [48] and text [59] with handheld controllers based or voice-based interactions for query input.

Ward et al. [55,56] focused on investigating search engine result pages in VR. They explored the balance between user comfort and task efficiency through varied spatial layouts of search results and three controller-based gesture interactions. Their later work [54] demonstrated a positional bias of the participants towards the top or top-left of the display and novel spatial layouts of the results had different navigational patterns from eye-tracking data.

Giunchi et al. [11] introduced a novel method for searching 3D model collections using free-form sketches with controllers within a virtual reality setting. This system was found to be more intuitive and efficient than traditional linear browsing methods [10]. Their later study showed that 3D mid-air sketching was the most intuitive and effective method compared to sketching on different 2D platforms.

ShapeFindAR [49] demonstrated a concept design of a dual search approach that combines spatial search queries, where users can sketch or trace in the physical environment, with textual search methods for 3D models.

2.2 Multimodal Interfaces in MR

Recent advancements in multimodal interfaces for MR focus on enhancing user interaction by integrating various input modalities like gaze, head movements, hand gestures, and other manual inputs [16,41]. GazeSwitch [15] leverages machine learning to optimize real-time switching between eye and head modes for hands-free pointing, offering a natural and adaptive interaction method, though it sacrifices some control and stability compared to manual switching. Similarly, PalmGazer [40] combines eye-hand interactions for a single-handed menu system in AR, facilitating quick digital commands through hand gestures and gaze, although tasks requiring higher degrees of freedom may still need two-handed input. Gaze-assisted text entry [31] reduces physical movement and eye fatigue by aligning gaze with manual input, outperforming gaze-only and manual-only methods. In addition, the Gaze-Hand Alignment technique [32] combines eye gaze and mid-air pointing to interact with menus in AR. This method improves the precision and speed of menu selections by aligning gaze with hand movements, reducing the cognitive load and physical effort required for interaction. Furthermore, gaze-shifting introduces a novel mechanism that modulates between direct and indirect inputs based on visual attention, applicable across pen and touch modalities to enhance dynamic interaction [39].

2.3 Context Switching

Mode-switching is critical for enhancing user interaction in HMDs, where traditional methods involve physical buttons [27], handheld controllers, and hand gestures [46,50]. These methods can be inefficient, especially when both hands are occupied. Alternative methods were introduced by recent studies to address the inefficiency. Several head gestures were explored for mode-switching in VR headsets [44]. Shi et al. [44] demonstrated the efficiency and efficacy of switching methods using four head gestures in a 3D painting application. HeadBoost [14] distinguished head gestures and head-gaze as different types of head movements and used head gestures for window switching in card sorting applications. Eye gaze was also introduced as a novel context-switching method. GazeDock [57] used a ring-shaped menu in the users' peripheral area of the field of view to trigger the menu automatically. Kuiper Belt [4] more specifically identified the angle region of 25° to 45° of the eye gaze, which could be leveraged as an input area to avoid the Midas touch problem.

2.4 Command Entry

Traditional menu-based interfaces are limited by their structure [2]: they show only a limited number of commands at once, necessitating hierarchical organization. This design forces users to navigate through multiple levels to reach a desired command, even if they know its location, and each hierarchical level introduces a potential for mode errors. In response to these limitations, various studies have investigated the use of gestural inputs as shortcuts for command execution on both desktop [6] and touchscreen devices [1,3,24,28,36]. This approach typically involves a predefined set of gestures, either mimicking the shape of specific commands or replicating command names in characters on a virtual keyboard. Users execute commands by tracing these patterns with their fingers on the touchscreen, aiming to improve both efficiency and user experience.

Command entry in MR is even more challenging due to the limited field of view for menu visualization and limited input capabilities. HoloBar [42] used a smartphone as a secondary modality to preview multi-level menus and validate command selections. OctoPocus in VR [8] extended the original dynamic guide on a touchscreen [3] into the context of Virtual Reality with 3D gestures. In a more recent exploration, HotGestures [47] combined command selection and control by employing a set of gestures in 3D modelling tasks.

Another way of bypassing a deep hierarchical menu apart from the gestural shortcuts is keyword search. Searching the commands by keywords allows users to directly access features or information by typing relevant keywords. Prior studies [29] have shown that users demonstrated the ability to complete 90% of web browsing tasks using keyword commands. This was later used for natural language interfaces in image editing [26,53,58] and programming [34]. MR technologies have expanded the scope of these applications by introducing novel ways of formulating and interacting with text queries [43].

2.5 Eye typing

Dwell-based eye typing, a longstanding research focus [12] for aiding individuals with physical disabilities, requires users to select keys by gazing at them for a specified duration [33]. Efforts to optimize this method, such as reducing dwell time and improving text prediction, have been limited by slow entry rates and eye strain. The concept of dwell-free eye typing [23] emerged as a solution, enabling text input by fixating on letters without a dwell threshold and using a statistical decoder for varied text granularities. This approach, theoretically capable of reaching high typing speeds, was later commercialized as a complementary product for disabled users [21]. Later studies [5,25,30,38] have mostly focused on word-at-a-time inputs, with systems employing unique mechanisms to reduce recognition errors and enhance efficiency.

3 SYSTEM DESIGN

In developing LookUP we sought to apply the six key design principles outlined below to guide the system design.

3.1 Design Principles

- **P1 Direct command access.** Implement search mechanisms that allow users to bypass traditional, multi-level menu navigation by allowing random access to commands. Use keyword-based command inputs to directly trigger functions, thereby simplifying interactions and reducing task completion times. This design principle leverages the limited field of view in

MR environments by avoiding an overloaded interface and minimizing the number of visible hierarchical layers.

- **P2 Visual clarity and workflow integration.** Maintain a clean and organized visual layout that avoids overcrowding the interface with excessive options. Design interactions that flow logically and intuitively from one step to the next, reducing the cognitive load by minimizing the need for users to recall or navigate through complex menu paths. This approach supports streamlined task execution and improves overall user experience.
- **P3 Enable a fast and accurate activation method while minimizing accidental activations.** Design the interface so that activation of tools is both quick and intentional, avoiding inadvertent triggers (addressing the Midas Touch problem [52]). Use specific, deliberate actions to activate the command search, ensuring that inputs are both fast and precise, enhancing the system's responsiveness and usability in dynamic environments.
- **P4 Reduce physical effort.** Opt for input methods that minimize physical strain, making the system suitable for prolonged use in various operational contexts.
- **P5 Facilitate occupied-hand scenarios.** Ensure the system design supports hands-free operation to accommodate scenarios where users' hands are engaged or when discretion is required. This is particularly valuable in professional settings or in situations where users must maintain attention on other tasks while interacting with the MR system.
- **P6 Flexible and scalable architecture.** As MR technologies evolve and the array of possible commands expands, design the system architecture to be flexible and scalable. The system should easily accommodate an increasing number of commands without necessitating redesigns.

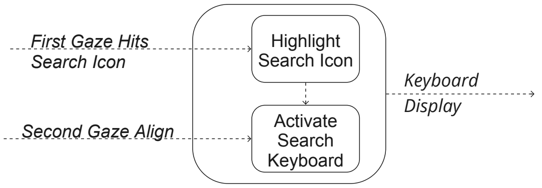
3.2 Activation

In the exploration of how to start command searching, we delineated the design space of switching the context from searching commands, focusing on the visual display location of the search icon and the selection mechanism used for its activation.

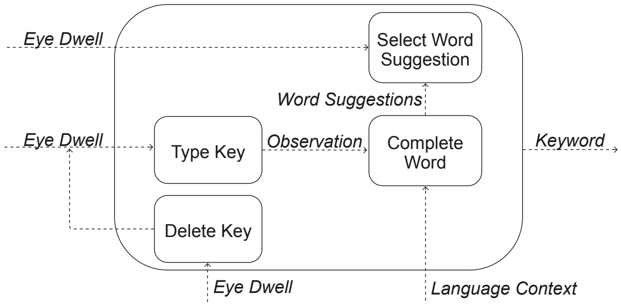
According to the design principles (P1, P3, P5), two novel activation techniques were distilled from the related work: Kuiper Belt [4] using eye gaze, and head and eye gaze alignment [45]. Both of the activation methods do not require dwell timeouts for selection and have been demonstrated to be efficient in prior studies. We decided to use head and eye gaze alignment as the HoloLens 2 is limited in its FoV and was not able to support users in looking at most of the peripheral regions from 25° to 45°.

We placed the search icon just above the user's field of view, as shown in Fig.1(a). This placement means that the icon does not obstruct the primary content, but is still quickly accessible when needed (P3). Prior research [18] also suggests that lower spatial positions tend to result in more occlusions, as the lower field often has a more complex environment than the upper one. Hence, we chose *top* rather than bottom or side positions.

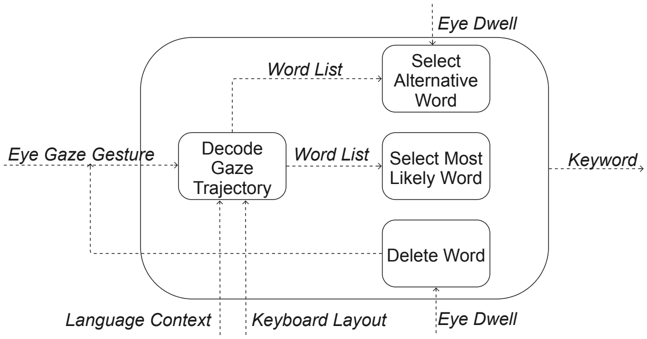
As shown in the function structure model in Fig. 2(a), when either of the head or eye gaze hits the search icon, the icon will be highlighted in a hover light. At that point, when the other gaze aligns within a 10° threshold, a virtual keyboard will be shown below the search icon. This means that the center of the user's field of view and the eye gaze point align on the search icon, to within 10°. Unlike state-of-the-art headsets like the Apple Vision Pro, which require users to visually search for a button and then confirm their selection with a pinch gesture or controller input, our method integrates head and eye gaze for both visual search and selection confirmation.



(a) Activation of search



(b) Dwell-based eye typing



(C) Dwell-free eye typing

Figure 2: Three function structures diagrams [19,20,22] for the search system design.

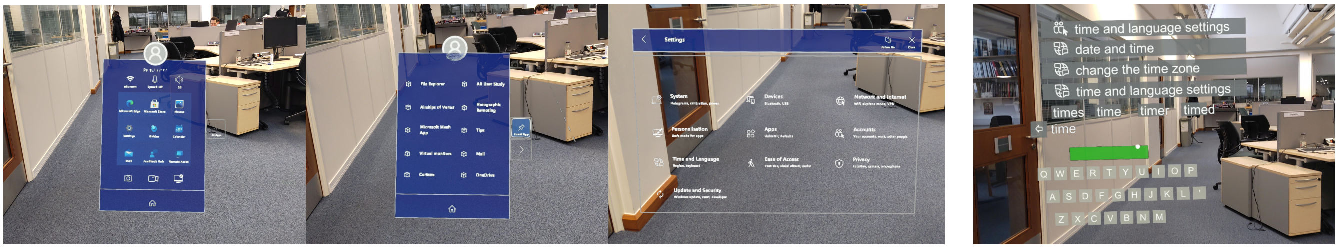
3.3 Command Search

3.3.1 Dwell-based eye typing

As a baseline condition, we implemented a standard dwell-based typing method. We subsequently refer to this condition as UpDwell. This method is based on a commercially available dwell-based keyboard [21] enhanced with word suggestion capabilities, as shown in Fig. 2(b). In this baseline condition, users are required to maintain their gaze on a key for a predetermined duration (500 ms). Once this dwell time is met, the character under the user's gaze is automatically selected by the system. To streamline the typing process and minimize the need for selecting each character individually, the keyboard displays a list of three suggested word completions, corresponding to the characters already typed. These suggestions appear at the top of the keyboard and can be chosen using the same dwell-based mechanism. The keyboard layout is identical to the UpDwellFree as shown in Figure 3 (d).

3.3.2 Dwell-free eye typing

To support command search by dwell-free eye typing, we adapted the keyboard introduced by Hu et al. [17], as shown in Fig. 2(c). We subsequently refer to this condition as UpDwellFree. This system initiates when the user gazes at the space bar, starting a



(a) (b) (c) (d)

Figure 3: The screenshots for the system from HoloLens 2. (a) The main menu (Level 1) replicates the native HoloLens 2 system menu. (b) 'All Apps' interface (Level 2). (c) 'Settings' interface (Level 2). Level 3 is identical to this interface but the user must go one layer deeper by pressing any of the buttons on the Level 2 interface. (d) The virtual keyboard for UpDwellFree. The user has just typed 'time' and the responses are shown at the top.

gaze trace that records the user's gaze sequence across the intended word's letters, concluding with a final gaze at the space bar. This process, analyzed using the I-VT algorithm [35], identifies fixations and saccades. The system then processes this data, focusing on the gaze trace from the first to the last fixation and discarding extraneous points. The user can drop any current gaze trace by looking outside of the keyboard area and retyping the word. The space key was positioned at the top of the virtual keyboard as shown in Figure 3 (d) for a smoother gaze flow which avoids the back-and-forth movements when checking results and word alternatives (P2).

A specially adapted gesture decoder processes these gaze points to suggest potential words. This gesture decoder employs a probabilistic model that accounts for letter sequence likelihood and key proximity and is capable of correcting common typing errors. This method allows for eye typing without the need for fixed dwell time on each key, providing a more fluid, sentence-level typing experience by linking word completion and initiation with a single gaze action on the space bar. The system has a vocabulary of 64,000 words. This can accommodate a growing array of commands without necessitating system redesigns, adhering to our system design principle (P6).

3.3.3 Hand Menu

We used the HoloLens 2's default holographic interface, with a particular focus on the Start menu, as a central calibration point. We subsequently refer to this condition as HandMenu. This Start menu interface, integral for user interaction and information retrieval within the HoloLens system, is typically activated by a gesture—tapping the Start icon on the user's wrist. Figure3 (a) shows the start menu in our simulation system.

In the middle part, the Start menu organizes applications into two categories: Pinned apps, which users can customize by adding or removing apps via a context menu that appears upon a prolonged press on an app tile, and All apps, which lists every application installed on the device, accessible through a button on the menu's right side.

We deactivated the standard gesture-based activation of the system's Start menu for our study to prevent the activation of the actual system menu. Then, we replicated the default gesture for bringing up the Start menu in our reproduced environment. We reproduced the default Start menu layout and behaviours to allow for use within our experimental application and allow for full logging of user interactions. This menu interface featured three hierarchical levels. The primary level comprised the Start menu and its internal buttons, accessible through a tap gesture. The secondary level became accessible upon interacting with the primary level buttons. For instance, selecting 'Settings' from the Pinned apps would reveal the settings list (Figure3 (a)), and pressing 'All apps' would display a complete list of available applications (see Figure 3 (b)). The third level went a layer further after the second level. The simulation reproduced the file system in HoloLens 2 and indexed all the applications and commands available.

3.4 Search Response Mechanism

Each time a new word was typed or a word alternative was selected, a shortlist of four commands would be shown in front of the user without occluding the keyboard view. We used the conventional bag-of-words to retrieve the available commands in the system for demonstration of the interaction technique. The icons on the left of the commands indicate the category of this searched item. For both the UpDwell and UpDwellFree conditions, the command was selected by 500 ms dwelling time. The deletion key on the left of the text input field permits character-level and word-level deletions for dwell-based and dwell-free methods respectively.

4 USER STUDY

We conducted a user study to evaluate the performance of the three command search and selection conditions: UpDwell, UpDwellFree and HandMenu.

4.1 Participants and apparatus

12 participants were recruited for the study, comprising 7 females and 5 males, aged between 20 and 36 years. Half of the participants ($n = 6$) reported prior experience with VR headsets, two were familiar with the HoloLens, and three had experience using eye-typing applications. As compensation for their participation, each individual received a voucher.

The study was conducted using a Microsoft HoloLens 2 headset, leveraging its built-in eye-tracking capability through the Extended Eye Tracking API. This feature captures eye gaze data at a rate of 90 Hz. The virtual keyboard was positioned 1.5 meters from the user, with dimensions of 0.27 meters in height and 0.7725 meters in width. Our system was developed using Unity and deployed directly onto the HoloLens 2 headset. All statistical decoding processes also take place on the headset itself.

4.2 Procedure

Participants were instructed to complete the three conditions within one hour. Before the study, participants completed a demographic questionnaire including questions about their previous experience with AR/VR headsets and familiarity with any eye-typing systems. At the beginning of the session, participants were first instructed to perform eye calibration for the device and were familiarised with the HoloLens 2. Subsequently, the three conditions were introduced by a video demonstration. The three conditions were fully balanced between the participants using a Latin Square. At the beginning of each condition, participants completed a short practice session. This practice sessions served to familiarize participants with the two typing methods and the menu system.

4.3 Task

In each condition, participants were asked to activate three groups of commands using the assigned interaction method, with each group containing the same nine commands. These commands were evenly distributed across the three hierarchical menu levels of the native HoloLens 2 interface and their order was randomized within each group. The term ‘Level’ here refers to the depth within the interface, starting from the high level (Level 1) and progressing to lower levels by navigating through subsequent layers to access specific options or information. Level 1 represents the primary Start menu, showcasing broader options and shortcuts to frequently used functions, such as ‘take a photo’. Level 2 delves one layer deeper, accessed through selections like ‘**All Apps**’ or ‘**Settings**’ from the main menu. Level 3 encompasses more detailed options found within the ‘**Settings**’ menu, consistent with those in other Windows operating systems. Overall, there are 12, 20 and 35 pressable buttons in the three respective levels. The nine commands chosen for this study were as follows:

- Level 1: ‘*Open* the **Photo** application’, ‘*Open* the **Mail** application’, ‘*Open* the **Remote Assist** application’.
- Level 2: ‘*Open* the **Tips** application’, ‘*Open* the **Virtual Monitors** application’, ‘*Open* **Ease of Access** menu’.
- Level 3: ‘*Open* **Environment** settings’, ‘*Open* **Country and Region** settings’, ‘*Open* **Start Gesture** settings’.

At the start of each command trial, the participants were first given the task instructions with a text display of the designated command. To indicate they were ready to commence the task, they either placed their hands inside a virtual cube in the native condition or kept their head gaze inside a virtual circle at the center of their field of view when they were at a neutral position in the other two gaze-based conditions. This initialization ensured a consistent starting position. Once their hands or head gaze left the starting area, the timer would start and they could perform the task with the designated method. As soon as the command was selected, the task was marked as complete. Participants were required to correct any errors and choose the target command before they could go to the next task. We recorded the completion time of the task and also recorded the selection time for typing the commands. They were also instructed to take a break before the next group. At the end of each condition, participants were asked to fill in NASA-TLX for workload assessment.

5 RESULTS

5.1 Completion Time

The completion time refers to the total time from activating the search function, or bringing up the menu, to completing the command selection. We utilized the completion time to assess the efficiency of the three methods, as depicted in Figure 4 (a). To analyze the total completion time, a two-way repeated measures ANOVA was conducted. The assumption of sphericity was not met, leading us to apply the Greenhouse-Geisser correction for adjustment. Our analysis revealed a significant main effect of Condition on the completion time ($F(1.33, 14.63) = 10.99$, $p < 0.05$, $\eta_p^2 = 0.32$) and a significant effect of Group ($F(1.47, 16.22) = 11.87$, $p < 0.05$, $\eta_p^2 = 0.09$). Subsequent pairwise comparisons indicated that the UPDWELLFREE method significantly outperformed the other two conditions in completion time with an average of 6.67 s which is 2.15 s and 1.90 s faster than HANDMENU and UPDWELL respectively. No significant difference in completion time was found between HANDMENU and UPDWELL. Further analysis showed that Group 3 required significantly less time than Groups 1 and 2, while no significant difference in completion time was observed between Groups 1 and 2. This result confirms that the performances of each method were improved through practice over the three repetitions. In the final group, the three methods HANDMENU, UPDWELL and UPDWELLFREE exhibited average completion times of 8.83 s, 7.34 s, and 6.14 s respectively.

We also observed a significant main effect of Menu Hierarchical Level on completion time ($F(1.33, 14.66) = 124.58$, $p < 0.05$, $\eta_p^2 = 0.23$) as shown in Figure 4 (b). Post-hoc analysis with Bonferroni correction revealed that the completion time required for commands from Levels 2 and 3 was significantly greater than that for Level 1. Additionally, selections at Level 3 took significantly longer than those at Level 2. Furthermore, we identified a significant interaction effect between CONDITION * MENULEVEL on completion time ($F(1.55, 17.02) = 71.99$, $p < 0.05$, $\eta_p^2 = 0.39$), as depicted in Figure 4 (b). This interaction indicates that the influence of Condition on completion time was dependent on MENULEVEL. Specifically, the post-hoc analysis revealed that the HANDMENU condition was significantly slower at lower levels compared to higher levels, whereas the UPDWELL and UPDWELLFREE conditions did not exhibit significant differences across levels. The completion times for HANDMENU in both Levels 1 and 2 and UPDWELLFREE methods for all three levels were statistically comparable. With Level 3, HANDMENU became significantly slower than both UPDWELL and UPDWELLFREE. UPDWELLFREE achieved the shortest mean completion time of 6.56 s in level 3 while HANDMENU had a mean of 13.75 s in the same level. This suggests that the efficiency of the UPDWELLFREE method, particularly in reducing completion time, is largely attributed to its performance with sub-menu items. The distinct mechanisms of UPDWELL and UPDWELLFREE, in contrast to HANDMENU, seem to neutralize the impact of menu levels by effectively flattening the hierarchical structure.

5.2 Selection Time

The definition of selection time in our study refers to the interval starting from when the virtual keyboard becomes visible upon activation and ending when the correct commands are successfully selected. Notably, this period constituted over 90% of the total completion time, on average, for the two typing conditions, UPDWELL and UPDWELLFREE. Due to the nature of these search techniques, the selection time is largely reliant on the text entry rate in typing. The assumption is that the selection time would be affected by the lengths of the keywords of each task. Figure 5 shows the selection times for these two typing conditions across the nine distinct tasks. A repeated measures ANOVA finds that UPDWELLFREE used significantly shorter selection time over UPDWELL ($F(1, 11) = 25.81$, $p < 0.05$, $\eta_p^2 = 0.37$) while the nine different tasks, which were sourced from three menu levels, did not significantly influence the selection time. This investigation reveals that the length of the keywords has no significant effect on the selection time. One reason could be that participants tend to only type one word even if there are two words in the task description. For instance, for **Ease of use**, as soon as ‘ease’ was typed, the command would be shown in response.

5.3 Deletions

The analysis of the number of deletions per task between the UPDWELL (M=0.318, SD=0.191) and UPDWELLFREE (M=0.361, SD=0.12) conditions revealed no significant difference, as shown in Fig.7. A paired samples t-test confirmed this lack of significant difference (t(11) = -0.723, p = 0.485). These results suggest that both interfaces exhibit similar error rates. Consequently, the significant difference in completion time between these two conditions is unlikely to be attributed to the number of error corrections per task. Given we only tested short commands consisting of 1 to 3 words,

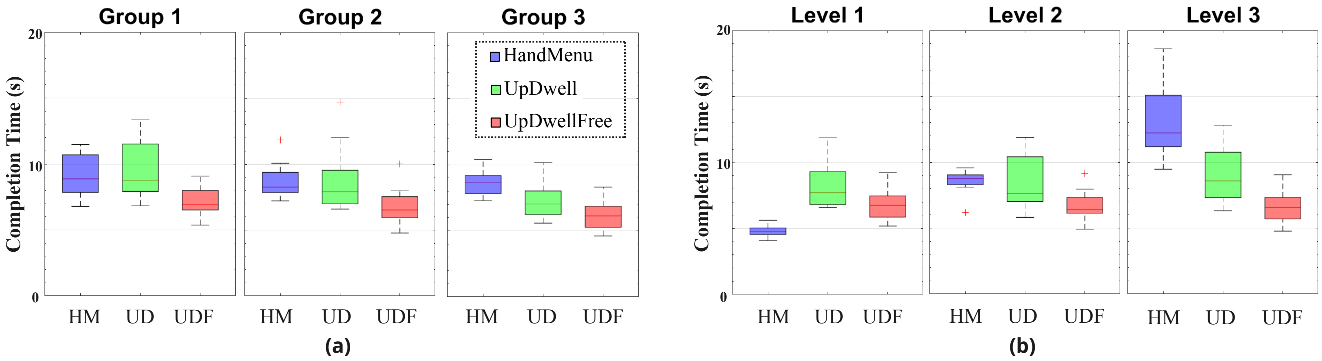


Figure 4: Boxplots in (a)(b) show the first quartile (Q1) and the third quartile (Q3) with a median represented by '-'. Whiskers show the minimum and maximum values. (a) The completion time of the three conditions HANDMENU (HM), UPDWELL (UD) and UPDWELLFREE (UDF) shown in box plots across three groups; (b) The completion time across tasks with three menu levels in box plots.

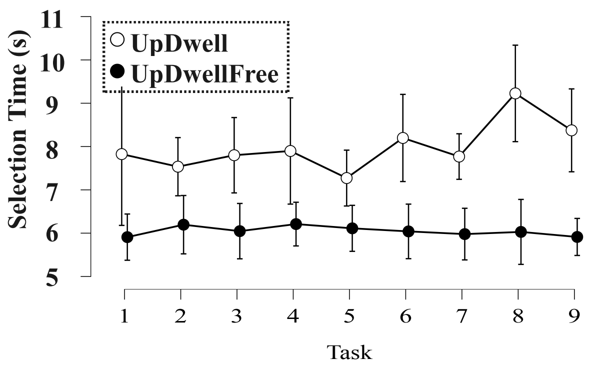


Figure 5: The selection time used for each task of the two typing conditions shown in line graph with error bars representing standard error. The '1' to '9' tasks represent **Start Gesture**, **Photo**, **Mail**, **Remote Assist**, **Tips**, **Virtual Monitors**, **Ease of Access**, **Environment**, **Country and Region** respectively.

further research with a larger sample size of commands or different performance metrics may be needed to explore other potential differences or benefits of these conditions.

5.4 Subjective Feedback

We measured the perceived workload using NASA-TLX subratings as shown in Figure 6. UPDWELLFREE was rated the lowest in physical demand and performance. Overall, no statistically significant difference was found on average ratings in workload between the three conditions.

6 DISCUSSION

Our user study revealed that the command search method using eye typing achieves a significantly shorter completion time by efficiently selecting the lower-level menu items at a similar speed as higher-level menu items while maintaining a comparable speed for selections on the main menu. Further, the dwell-free eye typing method achieved a significantly faster selection time than the conventional dwell-based method, further reducing the completion time for search tasks. With practice, participants achieved 30.5% shorter mean completion times in the UPDWELLFREE condition compared to our replication of the native menu system on the HoloLens 2, which served as the baseline.

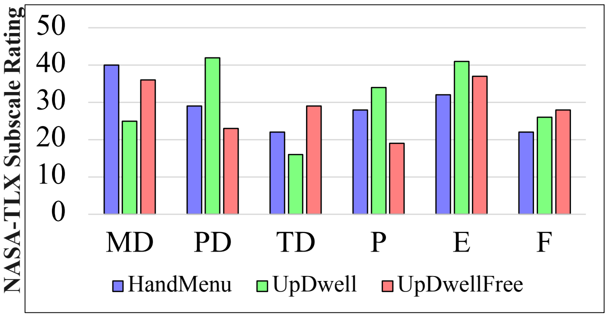


Figure 6: NASA-TLX results (Mental Demand (MD), Physical Demand (PD), Temporal Demand (TD), Performance (P), Effort (E) and Frustration (F)).

While we achieved a significant reduction in average completion time, the completion time associated with the HANDMENU increases exponentially as the hierarchy within the menu expands. In level 3, the UPDWELLFREE achieved a 52.3% reduction in completion time compared to the HANDMENU method. Looking at the boxplot of HANDMENU in level 3, the completion times were not only significantly longer than the other two levels, but also exhibited a greater variability in the data. It is expected that both interaction and exploration times increase with the addition of more menu layers. This exponential increase may be attributed to the cognitive load and navigation complexity introduced by deeper menu structures. On the other hand, for command search methods, the hierarchy of the menu does not impact completion time, remaining consistent regardless of the number of layers. This stability suggests that command search methods bypass the cognitive and navigational challenges presented by hierarchical menus.

To broaden the scope and utility of hands-free command search systems in mixed reality applications, future work should focus on incorporating a wider range of content types. This includes not only text, but also images, 3D objects, videos, and contextual data. By textualizing or tokenizing these varied content forms, the system can be adapted to search across a broader spectrum of elements, enhancing the system's versatility.

Beyond the applications on the main system menu, our command search approach could potentially benefit creative work, accessibility, and industrial settings. In creative contexts like sketching and 3D modelling, the ability to switch tools rapidly using eye movements

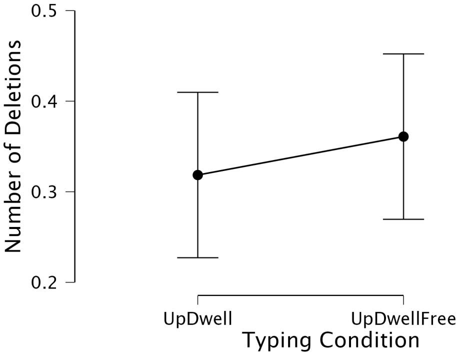


Figure 7: The number of deletions per task used for each task of the two typing conditions. The error bars indicate the standard error rates.

enhances efficiency by allowing artists and designers to change brushes, colors, or perspectives seamlessly without interrupting their workflow. For accessibility, the system empowers users with motor impairments to control applications and tools independently through eye movements, potentially combined with voice commands for a comprehensive hands-free experience. In industrial and field settings, where workers need to keep their hands free, the LookUP system can improve safety and efficiency by enabling technicians to access schematics, adjust machine settings, or log data using eye-typing, which is particularly advantageous in hygiene-critical environments, such as medical or food processing industries.

The proposed activation mechanism of the LookUP technique could be further enhanced by integrating it with other input modalities and UI elements to enable fast shortcuts and context switching. For example, frequently used functions or applications could appear as shortcuts after activating the search bar. Additionally, the ”look up” mechanism could serve as a window switch shortcut or facilitate context switches between the immersive digital environment and the physical world, enhancing user experience and operational efficiency across various applications.

Additionally, contextual information could be leveraged as an additional input for command search systems in MR. This would allow the system to tailor searches and responses based on the user’s environment or task at hand, improving relevance and efficiency. By dynamically integrating context, such as the user’s location, the objects they are interacting with, or their calendar details, the system can offer more precise and actionable information, enhancing the overall user experience.

We also acknowledge that recruiting a broader demographic representation would ensure findings are more generalizable and accessible to diverse users in our future studies. It would also be interesting to investigate the impact of user characteristics such as age, vision quality, and VR experience on system performance.

7 CONCLUSION

In summary, in this work, we presented LookUP, an effective and efficient hands-free command search system for mixed reality systems that leverages the dwell-free eye typing paradigm to allow users to input commands when encumbered or when their hands are not readily available. We demonstrated the feasibility and superior efficiency of the system by comparing it with a baseline system consisting of a default menu-based system using hand touch input, as well as a dwell-based eye typing baseline method. Overall, LookUP resulted in faster selection and completion times compared to the prevailing de facto established command entry solution. We hope this work will inspire further explorations of leveraging dwell-free eye typing mechanisms in mixed reality.

ACKNOWLEDGMENTS

John Dudley and Per Ola Kristensson were supported by EPSRC (grants EP/S027432/1 and EP/W02456X/1).

REFERENCES

[1] J. Alvina, C. F. Griggio, X. Bi, and W. E. Mackay. Commandboard: Creating a general-purpose command gesture input space for soft keyboard. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology*, pp. 17–28, 2017.

[2] G. Bailly and S. Malacria. Command selection. In *Handbook of Human Computer Interaction*, pp. 1–35. Springer, 2022.

[3] O. Bau and W. E. Mackay. Octopocus: a dynamic guide for learning gesture-based command sets. In *Proceedings of the 21st annual ACM symposium on User interface software and technology*, pp. 37–46, 2008.

[4] M. Choi, D. Sakamoto, and T. Ono. Kuiper belt: Utilizing the “out-of-natural angle” region in the eye-gaze interaction for virtual reality. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*, pp. 1–17, 2022.

[5] W. Cui, R. Liu, Z. Li, Y. Wang, A. Wang, X. Zhao, S. Rashidian, F. Baig, I. Ramakrishnan, F. Wang, et al. Glancewriter: Writing text by glancing over letters with gaze. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, pp. 1–13, 2023.

[6] W. Cui, J. Zheng, B. Lewis, D. Vogel, and X. Bi. Hotstrokes: Word-gesture shortcuts on a trackpad. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, pp. 1–13, 2019.

[7] R. Dachselt and A. Hübner. A survey and taxonomy of 3d menu techniques. In *EGVE*, vol. 6, pp. 89–99, 2006.

[8] K. Fennedy, J. Hartmann, Q. Roy, S. T. Perrault, and D. Vogel. Octopocus in vr: using a dynamic guide for 3d mid-air gestures in virtual reality. *IEEE Transactions on Visualization and Computer Graphics*, 27(12):4425–4438, 2021.

[9] B. Fruchard, E. Lecolinet, and O. Chapuis. Side-crossing menus: enabling large sets of gestures for small surfaces. *Proceedings of the ACM on Human-Computer Interaction*, 4(ISS):1–19, 2020.

[10] D. Giunchi, S. James, and A. Steed. 3d sketching for interactive model retrieval in virtual reality. In *Proceedings of the Joint Symposium on Computational Aesthetics and Sketch-Based Interfaces and Modeling and Non-Photorealistic Animation and Rendering*, pp. 1–12, 2018.

[11] D. Giunchi, S. James, and A. Steed. Model retrieval by 3d sketching in immersive virtual reality. In *2018 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 559–560. IEEE, 2018.

[12] A. Hamid and P. O. Kristensson. 40 years of eye typing: Challenges, gaps, and emergent strategies. *Proc. ACM Hum.-Comput. Interact.*, 8(ETRA), may 2024. doi: 10.1145/3655596

[13] D. Henriques, D. Mendes, P. Pascoal, I. Trancoso, and A. Ferreira. Poster: Evaluation of immersive visualization techniques for 3d object retrieval. In *2014 IEEE Symposium on 3D User Interfaces (3DUI)*, pp. 145–146. IEEE, 2014.

[14] B. J. Hou, J. Newn, L. Sidenmark, A. Ahmad Khan, P. Bækgaard, and H. Gellersen. Classifying head movements to separate head-gaze and head gestures as distinct modes of input. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, pp. 1–14, 2023.

[15] B. J. Hou, J. Newn, L. Sidenmark, A. A. Khan, and H. Gellersen. Gazeswitch: Automatic eye-head mode switching for optimised hands-free pointing. *Proc. ACM Hum.-Comput. Interact.*, 8(ETRA), may 2024. doi: 10.1145/3655601

[16] J. Hu, J. J. Dudley, and P. O. Kristensson. An evaluation of caret navigation methods for text editing in augmented reality. In *2022 IEEE International Symposium on Mixed and Augmented Reality Adjunct (ISMAR-Adjunct)*, pp. 640–645, 2022. doi: 10.1109/ISMAR-Adjunct57072.2022.00132

[17] J. Hu, J. J. Dudley, and P. O. Kristensson. Skimr: Dwell-free eye typing in mixed reality. In *2024 IEEE Conference Virtual Reality and 3D User Interfaces (VR)*, pp. 439–449. IEEE, 2024.

[18] E. M. Klose, N. A. Mack, J. Hegenberg, and L. Schmidt. Text presentation for augmented reality applications in dual-task situations. In *2019*

IEEE Conference on Virtual Reality and 3D User Interfaces (VR), pp. 636–644, 2019. doi: 10.1109/VR.2019.8797992

[19] P. O. Kristensson. Designing virtual and augmented reality user interfaces using parameterized function structure models. In *2024 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)*, pp. 419–423. IEEE, 2024.

[20] P. O. Kristensson, J. Lilley, R. Black, and A. Waller. A design engineering approach for quantitatively exploring context-aware sentence retrieval for nonspeaking individuals with motor disabilities. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, pp. 1–11, 2020.

[21] P. O. Kristensson, M. Mjelde, and K. Vertanen. Understanding adoption barriers to dwell-free eye-typing: Design implications from a qualitative deployment study and computational simulations. In *Proceedings of the 28th International Conference on Intelligent User Interfaces*, pp. 607–620, 2023.

[22] P. O. Kristensson and T. Müllners. Design and analysis of intelligent text entry systems with function structure models and envelope analysis. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, pp. 1–12, 2021.

[23] P. O. Kristensson and K. Vertanen. The potential of dwell-free eye-typing for fast assistive gaze communication. In *Proceedings of the symposium on eye tracking research and applications*, pp. 241–244, 2012.

[24] P. O. Kristensson and S. Zhai. Command strokes with and without preview: using pen gestures on keyboard for command selection. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 1137–1146, 2007.

[25] A. Kurauchi, W. Feng, A. Joshi, C. Morimoto, and M. Betke. Eyeswipe: Dwell-free text entry using gaze paths. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, pp. 1952–1956, 2016.

[26] G. P. Laput, M. Dontcheva, G. Wilensky, W. Chang, A. Agarwala, J. Linder, and E. Adar. Pixeltone: A multimodal interface for image editing. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 2185–2194, 2013.

[27] J. J. LaViola Jr, E. Kruijff, R. P. McMahan, D. Bowman, and I. P. Poupyrev. *3D user interfaces: theory and practice*. Addison-Wesley Professional, 2017.

[28] H. V. Le, S. Mayer, M. Weiß, J. Vogelsang, H. Weingärtner, and N. Henze. Shortcut gestures for mobile text editing on fully touch sensitive smartphones. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 27(5):1–38, 2020.

[29] G. Little and R. C. Miller. Translating keyword commands into executable code. In *Proceedings of the 19th annual ACM symposium on User interface software and technology*, pp. 135–144, 2006.

[30] Y. Liu, C. Zhang, C. Lee, B.-S. Lee, and A. Q. Chen. Gazetry: Swipe text typing using gaze. In *Proceedings of the annual meeting of the australian special interest group for computer human interaction*, pp. 192–196, 2015.

[31] M. N. Lystbæk, K. Pfeuffer, J. E. S. Grønbæk, and H. Gellersen. Exploring gaze for assisting freehand selection-based text entry in ar. *Proc. ACM Hum.-Comput. Interact.*, 6(ETRA), may 2022. doi: 10.1145/3530882

[32] M. N. Lystbæk, P. Rosenberg, K. Pfeuffer, J. E. Grønbæk, and H. Gellersen. Gaze-hand alignment: Combining eye gaze and mid-air pointing for interacting with menus in augmented reality. *Proc. ACM Hum.-Comput. Interact.*, 6(ETRA), may 2022. doi: 10.1145/3530886

[33] P. Majaranta and K.-J. Räihä. Twenty years of eye typing: systems and design issues. In *Proceedings of the 2002 symposium on Eye tracking research & applications*, pp. 15–22, 2002.

[34] T. Nguyen, P. C. Rigby, A. T. Nguyen, M. Karanfil, and T. N. Nguyen. T2api: Synthesizing api code usage templates from english texts with statistical translation. In *Proceedings of the 2016 24th ACM SIGSOFT International Symposium on Foundations of Software Engineering*, pp. 1013–1017, 2016.

[35] A. Olsen. The tobii i-vt fixation filter. *Tobii Technology*, 21:4–19, 2012.

[36] T. Ouyang and Y. Li. Bootstrapping personal gesture shortcuts with the wisdom of the crowd and handwriting recognition. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 2895–2904, 2012.

[37] C. Park, H. Cho, S. Park, Y.-S. Yoon, and S.-U. Jung. Handposemenu: Hand posture-based virtual menus for changing interaction mode in 3d space. In *Proceedings of the 2019 ACM international conference on interactive surfaces and spaces*, pp. 361–366, 2019.

[38] D. Pedrosa, M. D. G. Pimentel, A. Wright, and K. N. Truong. Filteryedping: Design challenges and user performance of dwell-free eye typing. *ACM Transactions on Accessible Computing (TACCESS)*, 6(1):1–37, 2015.

[39] K. Pfeuffer, J. Alexander, M. K. Chong, Y. Zhang, and H. Gellersen. Gaze-shifting: Direct-indirect input with pen and touch modulated by gaze. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology*, UIST '15, p. 373–383. Association for Computing Machinery, New York, NY, USA, 2015. doi: 10.1145/2807442.2807460

[40] K. Pfeuffer, J. Obernolte, F. Dietz, V. Mäkelä, L. Sidenmark, P. Manakhov, M. Pakanen, and F. Alt. Palmgazer: Unimanual eye-hand menus in augmented reality. In *Proceedings of the 2023 ACM Symposium on Spatial User Interaction*, SUI '23. Association for Computing Machinery, New York, NY, USA, 2023. doi: 10.1145/3607822.3614523

[41] I. Rakkolainen, A. Farooq, J. Kangas, J. Hakulinen, J. Rantala, M. Turunen, and R. Raisamo. Technologies for multimodal interaction in extended reality—a scoping review. *Multimodal Technologies and Interaction*, 5(12):81, 2021.

[42] H. Saidi, E. Dubois, and M. Serrano. Holobar: Rapid command execution for head-worn ar exploiting around the field-of-view interaction. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, pp. 1–17, 2021.

[43] M. Schleußinger, P. Hansen, and R. Ramberg. Immersive information seeking–a scoping review of information seeking in virtual reality environments. *Journal of Information Science*, p. 01655515231174384, 2023.

[44] R. Shi, N. Zhu, H.-N. Liang, and S. Zhao. Exploring head-based mode-switching in virtual reality. In *2021 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pp. 118–127. IEEE, 2021.

[45] L. Sidenmark and H. Gellersen. Eye&head: Synergetic eye and head movement for gaze pointing and selection. In *Proceedings of the 32nd annual ACM symposium on user interface software and technology*, pp. 1161–1174, 2019.

[46] Z. Song, J. J. Dudley, and P. O. Kristensson. Efficient special character entry on a virtual keyboard by hand gesture-based mode switching. In *2022 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pp. 864–871. IEEE, 2022.

[47] Z. Song, J. J. Dudley, and P. O. Kristensson. Hotgestures: Complementing command selection and use with delimiter-free gesture-based shortcuts in virtual reality. *IEEE Transactions on Visualization and Computer Graphics*, 2023.

[48] F. Spiess, R. Gasser, S. Heller, L. Rossetto, L. Sauter, and H. Schuldt. Competitive interactive video retrieval in virtual reality with vitrivr-vr. In *International Conference on Multimedia Modeling*, pp. 441–447. Springer, 2021.

[49] E. Stemasov, T. Wagner, J. Gugenheimer, and E. Rukzio. Shapefindar: Exploring in-situ spatial search for physical artifact retrieval using mixed reality. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*, pp. 1–12, 2022.

[50] H. B. Surale, F. Matulic, and D. Vogel. Experimental analysis of barehand mid-air mode-switching techniques in virtual reality. In *Proceedings of the 2019 CHI conference on human factors in computing systems*, pp. 1–14, 2019.

[51] T. Teo, M. Norman, M. Adcock, and B. H. Thomas. Data fragment: Virtual reality for viewing and querying large image sets. In *2017 IEEE Virtual Reality (VR)*, pp. 327–328. IEEE, 2017.

[52] B. Velichkovsky, A. Sprenger, and P. Unema. Towards gaze-mediated interaction: Collecting solutions of the "midas touch problem". In *Human-Computer Interaction INTERACT'97: IFIP TC13 International Conference on Human-Computer Interaction, 14th–18th July 1997, Sydney, Australia*, pp. 509–516. Springer, 1997.

[53] Y. Wang, Z. Hou, L. Shen, T. Wu, J. Wang, H. Huang, H. Zhang, and D. Zhang. Towards natural language-based visualization author-

ing. *IEEE Transactions on Visualization and Computer Graphics*, 29(1):1222–1232, 2022.

[54] A. Ward, B. Choi, and R. Capra. Eyes on immersive search: Eye-tracking study of search engine result pages in immersive virtual environments. In *Proceedings of the 2023 Conference on Human Information Interaction and Retrieval*, pp. 43–54, 2023.

[55] A. R. Ward and R. Capra. Immersive search: Using virtual reality to examine how a third dimension impacts the searching process. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 1621–1624, 2020.

[56] A. R. Ward, Y. Gu, S. Avula, and P. Chakravarthula. Interacting with information in immersive virtual environments. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 2600–2604, 2021.

[57] X. Yi, Y. Lu, Z. Cai, Z. Wu, Y. Wang, and Y. Shi. Gazedock: Gaze-only menu selection in virtual reality using auto-triggering peripheral menu. In *2022 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 832–842. IEEE, 2022.

[58] T. Zhang, H.-Y. Tseng, L. Jiang, W. Yang, H. Lee, and I. Essa. Text as neural operator: Image manipulation by text instruction. In *Proceedings of the 29th ACM International Conference on Multimedia*, pp. 1893–1902, 2021.

[59] A. J. Zhou and G. H. Yang. Minority report by lemur: supporting search engine with virtual reality. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, pp. 1329–1332, 2018.