"Where did my apps go?" Supporting Scalable and Transition-Aware Access to Everyday Applications in Head-Worn Augmented Reality

Feiyu Lu, Leonardo Pavanatto, Shakiba Davari, Lei Zhang, Lee Lisle, Doug A. Bowman

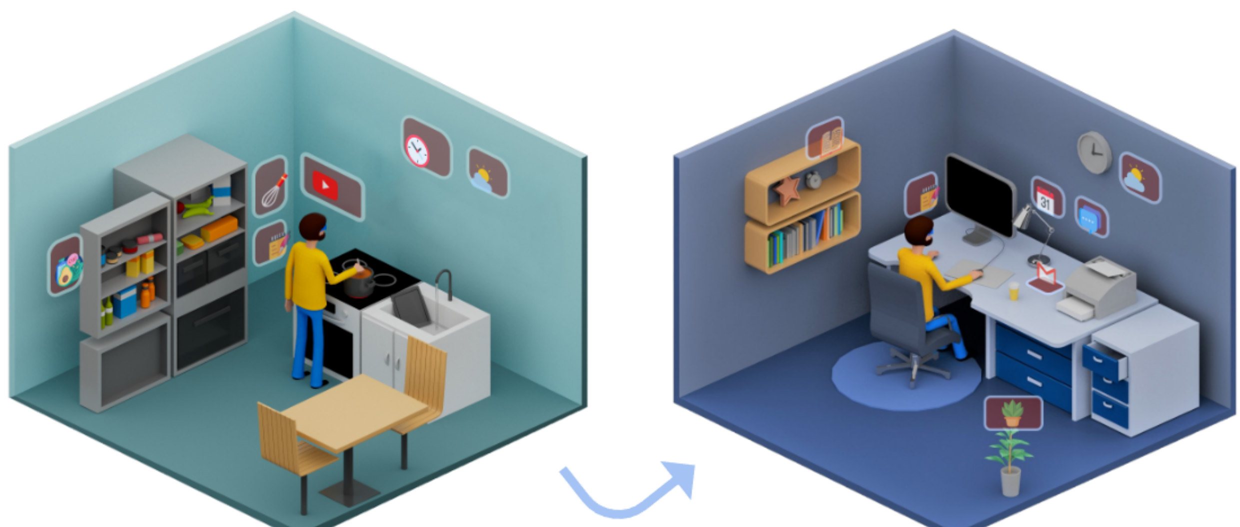


Fig. 1. An illustration of a transition-aware interface, in which the system automatically brings a subset of all AR applications that are relevant to the user's task space after the user transitions to a new space to seamlessly support their task activities. Left: in the kitchen environment, the Recipe, Fitness, Notes, Clock, Weather, and Video apps were opened; Right: when the user moves to the office, the system brings the Note and Weather app, and automatically opens the Email, Calendar, Message apps to be ready to assist with productivity tasks, as well as the Reading List and Plant apps that are relevant to the office environment.

Abstract— Future augmented reality (AR) glasses empower users to view personal applications and services anytime and anywhere without being restricted by physical locations and the availability of physical screens. In typical everyday activities, people move around to carry out different tasks and need a variety of information on the go. Existing interfaces in AR do not support these use cases well, especially when the number of applications increases. We explore the usability of three world-referenced approaches that move AR applications with users as they transition among different locations, featuring different levels of AR app availability: (1) always using a menu to manually open an app when needed; (2) automatically suggesting a relevant subset of all apps; and (3) carrying all apps with the users to the new location. Through a controlled study and a relatively more ecologically-valid study in AR, we reached better understandings on the performance trade-offs and observed the impact of various everyday contextual factors on these interfaces in more realistic AR settings. Our results shed light on how to better support the mobile information needs of users in everyday life in future AR interfaces.

Index Terms—Augmented reality, adaptive interface, glanceable interface, automation, mobile computing

1 Introduction

Augmented reality (AR) devices enable people to interact with their physical surroundings and the digital augmentations displayed on top [5]. With the recent advancements in hardware, AR head-worn displays (HWDs), or AR glasses, are becoming increasingly lightweight and portable. Many believe that AR will replace mobile phones as the next-generation mobile computing platform, empowering everyday users to view and interact with digital information anywhere and anytime regardless of the availability of physical screens [8, 30, 44].

In typical everyday situations, people move around in space, carry out different tasks, and need a variety of information on the go. The needs for information are mostly prompted by concurrent activities, environments, and location changes [23, 64]. One of the major benefits that mobile phones bring to people's lives is their ability to fulfill such information needs conveniently. However, current AR interfaces are not as efficient in handling these situations. Regarding existing state-of-the-art AR operating systems (e.g., the Magic Leap One and the HoloLens 2), AR content defaults to staying at a fixed location until users manually move or re-instantiate it. To enhance the mobility of

- *Feiyu Lu, Leonardo Pavanatto, Shakiba Davari, Lee Lisle, Doug A. Bowman are with Center for Human-Computer Interaction, Department of Computer Science, Virginia Tech, Blacksburg, VA, United States. E-mail: {feiyulu|lpavanat|sdavari|llisle|dbowman}@vt.edu*
- *Lei Zhang is with College of Computing and Software Engineering, Kennesaw State University, Kennesaw, GA, United States. E-mail: lzhang24@kennesaw.edu*

Manuscript received xx xxx. 201x; accepted xx xxx. 201x. Date of Publication xx xxx. 201x; date of current version xx xxx. 201x. For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org. Digital Object Identifier: xx.xxxx/TVCG.201x.xxxxxxx

AR content, HoloLens 2 provides a tag-along interface for the AR content to loosely follow the user's body. However, it requires the users to manually enable it, supporting only one piece of AR content at a time in its full size for following the user, with the risk of being distracting and obtrusive. These limitations assume that the main use cases for AR are confined to a single space, limiting the mobility and accessibility of digital content when users move around spatially from one location to another. After entering a different environment, users need to find and open/reopen the applications they need through a universal menu interface, which could be cumbersome and time-consuming in demanding everyday scenarios where hands are often already occupied.

As such, recent research has explored the possibility of carrying AR content with users while moving. Research has explored display-referenced and body-referenced layouts to fix the AR content for users to enhance their mobility [27, 48, 54, 55]. The major limitation of these approaches is ***scalability*** (i.e., when the number of applications in use increases, it can be cumbersome and obtrusive to have all the applications follow the user everywhere). According to available statistics, an average mobile phone owner in the US has more than 80 apps downloaded, uses 9-10 apps per day, and 30 apps per month [1, 2]. Similarly, it is likely that an everyday AR user would need access to a moderately large number of different virtual applications.

Information needs change frequently depending on locations and task activities. For example, the users may need the *timer*, *recipe*, and the *video streaming* apps in the kitchen to get ready for cooking, and need the *email*, *calendar*, *instant messaging*, and *to-do list* apps in the office to conduct productivity work. Due to the system's lack of knowledge about such dynamic needs of the user, access to information is provided either through the user's manual interaction or through the constant display of all the AR content, offloading the need to open any app manually. However, both approaches can pose an extra workload on the users. In manual interaction, as demonstrated in current systems, the user needs to manually search for and reopen the needed applications every time they are needed in a new location, which implies low availability of information and a high physical or mental workload. Constant display of all the AR content can ensure high availability of information, but at the cost of inducing information overload, distraction, and unwanted occlusions of the real-world environment, especially with the scaling of the number of AR applications.

One alternative could be that we partially offload the user's input to the system [31, 56]. In recent mobile phone interfaces (e.g., Apple iOS's contextual suggestions), the system suggests which applications to open based on location, time of day, and the user's interaction history. Similarly, an AR system could intelligently surface a subset of the total applications with a higher probability of being needed at the new location. Not only could this approach alleviate the information overload issue by having a smaller number of applications open, but also the prediction made by AR displays could theoretically lead to a higher level of accuracy than current mobile devices given the hardware's rich sensor capabilities [3, 22, 37, 38, 56, 62]. Recent SDKs such as ARKit RoomPlan [1] and Quest Scene API [2] provides support for automatic mapping of indoor room layout with limited amount of manual input. These information could then be retrieved by other applications after an initial setup. By inferring the user's tasks and upcoming destinations, the AR display could surface the corresponding applications in advance, being ready to accompany the tasks and assist the users seamlessly after their arrival. We acknowledge that accurately predicting user intent remains an ongoing challenge in the research community, though recent work has shed light on such possibilities with the advancements of multimodal AI [25, 49]. Our work explores the usability of such approaches, by having relevant information automatically appearing at the destination of user transitions.

In this work, we investigated how to enable the efficient acquisition of information in AR applications when users transition across multiple spatial locations, taking into account ***availability*** and ***scalability***. Specifically, we explored three interface designs after the user transitions spatially from location A to location B: (1) *None*: all previously opened apps disappear, and the user needs to manually (re-)open apps through a menu, similar to existing solutions; (2) *Some*: the system intelligently opens some relevant applications in location B, with the possibility of needing to use the menu if the needed app is not automatically opened; and (3) *All*: the system automatically opens all the applications in the new space. Through two user studies simulating how everyday users transition across multiple spaces and need information on-the-go, we investigated the trade-offs among the three interfaces both objectively and subjectively by answering three questions: (1) How is user performance of accessing information impacted when availability and the number of total apps increase? (2) What are the perceived agency, usability, and workload of the interfaces? (3) How do different contextual factors influence the user's preference of these interfaces? Our study is one of the first to explore the scalability and usability of AR interfaces through ecologically valid simulation of in-situ transitions.

The main contributions of this work are five-fold: (1) design explorations of three interfaces that support the user's mobile information needs during spatial transitions; (2) an initial exploration of user performance trade-offs of these interfaces when the number of AR applications scales through a controlled simulation study; (3) investigations of trade-offs among these interfaces in different contexts; (4) evaluation of user preference through a more ecologically-valid AR study; and (5) discussions of our unique lessons learned while taking into consideration user performance, usability, workload, preference, and usage patterns in comparison to prior work.

2 Related Work

2.1 Everyday AR and mobile information needs

With the advancements in hardware, recent AR/MR devices are becoming increasingly lightweight and are reaching the everyday consumer market for productivity and entertainment purposes[3] [4]. However, developments in interface and interactions seem to be lagging behind, with most existing solutions focusing primarily on uses in a single confined static space. However, users are often on the move while needing access to digital services concurrently [17, 18, 41, 64]. For example, Church & Smyth found that *on-the-go* was the most frequent location context where information needs arise [18]. Sohn et al. found that 55% of the user's information needs on-the-go were addressed late or left unaddressed. 72% of the overall information needs were prompted by locations, tasks, and activities of the users, in which location was the most prominent trigger (34.6%) [64].

As such, research has investigated interfaces that are more lightweight and mobile-friendly in order to support on-the-go access to AR content [27, 29, 48, 51, 54, 55]. For example, Lages & Bowman looked into adapting AR windows from world-anchored to body-anchored for access during walking [48]. Lu and colleagues explored display and body-referenced metaphors for convenient access to AR applications in everyday scenarios [54, 55]. Pfeuffer et al. proposed Gaze+Pinch, a lightweight multimodal interface for everyday interactions with UIs in AR/VR [60]. These work shed light on *adaptable* UIs to assist with continuous interactions with AR apps on-the-go.

Although *adaptable* UI offers more user agency and consistency, it demands user feedback sporadically, which may be nonideal in heavy and dynamic scenarios. Motivated by this, research also explored *adaptive* interfaces which spontaneously react to contextual changes. Early work in view management explores adaptive layout of 2D labels in users' viewport to prevent occlusion and cluttering using optimization-based methods [7, 35, 46]. Tatzgern et al. proposed adaptable information density, in which groups of AR labels fold and unfold themselves based on user preference and interactions [68]. Lindlbauer et al. explored AR interfaces that automatically adjust level of detail and frame of reference of AR content for minimizing distractions and offering task assistance [51]. Belo et al. deployed a adaptive UI toolkit for optimizing

[1] https://developer.apple.com/augmented-reality/roomplan/
[2] https://developer.oculus.com/documentation/unity/unity-scene-overview/

[3] Meta Quest Pro: https://www.meta.com/quest/quest-pro/
[4] Vive XR Elite: https://www.vive.com/us/product/vive-xr-elite/overview/

the placement of 3D UIs according to a set of objectives [28]. Additionally, Diverdi et al. proposed level of detail interface, which adapts its level of information displayed based on the user's distance to it [24]. Gebhardt proposed a model-free reinforcement-learning method for adapting the visibility of AR labels in virtual environment [34]. These work shed light on how visibility of AR content should be automatically adapted to support rather than overwhelming the users.

In this work, we focus on *adaptive* and *world-anchored* UIs for mobile-friendly information access in AR, which has been covered by limited prior work. By situating content in 3D physical locations, they could interfere with the user's view less [50], possess spatial consistency [26], and leverage the semantics of the environments and objects in the physical scene to memorize and make sense of where and why a piece of AR content was placed [14, 57]. In this study, we focus primarily on such possibilities, by exploring scalability and usability of world-anchored approaches to adapting AR interfaces for on-the-go access.

2.2 Environmentally-adaptive MR interfaces

Advancements in computer vision and tracking enable recent AR/MR devices to possess a decent level of scene understanding. For example, Microsoft HoloLens and Magic Leap construct a spatial mesh in real time, the data of which is accessible through API calls. Apple's ARKit also recently released their RoomPlan API for iOS, which allows an AR app to generate a floor plan for the room after a quick scan[5]. These developments lower the barrier for developers to access environmental data and lead to huge potential for future AR/VR interfaces to leverage the dynamic physical environments that surround users to place AR content for continuous, rapid, and convenient acquisitions.

In earlier work, Gal et al. explored algorithms for rapid deployment of AR content in a physical environment taking into account the scene geometry [33]. Nuernberger et al. explored techniques that enable precise positioning of AR content while considering the characteristics and constraints of physical scenes [58]. Sra et al. explored procedural generation of virtual environments according to the scanning of the user's physical environment [65]. Lindlbauer and Wilson proposed a system that utilized 3D depth captures to recreate a physical room in MR [52].

These projects focused on adaptation of virtual elements to a single physical environment. More recently, Ens et al. explored ways of laying out AR windows across multiple physical environments consistently across space while maintaining their spatial relationships [26]. Caetano and Sra proposed ARfy, a pipeline for adapting AR content to multiple physical scenes without requiring user intervention and annotation [11]. Han et al. studied blending AR windows on top of physical objects in any environment using optimization-based methods [39]. Cheng et al. proposed layout adaptations of MR interfaces according to the semantics of multiple scenes and the affordance of physical surfaces for interactions [14, 15]. Lu & Xu studied automated AR interfaces to support user transition across multiple environments, which we consider the most relevant to our work. Although the work also researched the usability of manual and automated interface solutions that support application acquisitions during user transitions, the study was conducted in a simulated setting similar to other prior works in the literature. There is lack of empirical evaluations of such approaches in a AR setup with when users actually physically transition across multiple physical environments, instead of in simulated VR environments. Moreover, questions about the user performance of accessing information from world-referenced AR content adapted across multiple physical environments, and user preference during different user tasks and activities, remain unanswered. In this work, we attempted to fill this gap by studying such approaches in relatively more realistic AR scenarios involving user transitions and tasks across multiple room spaces in a large physical environment.

[5]https://developer.apple.com/augmented-reality/roomplan/

2.3 Performance modeling and scalability

Modeling user performance is critical for developers to understand how different factors impact task completion time and make the best design choices in order to maximize efficiency. Notable user performance models in HCI include Fitts's Law [32] and KLM/GOMS [12]. Similar models have been extended to phones [42], smartwatches [4], and in VR/AR [10, 16, 47]. There has been extensive work or modelling user performance in the context of traditional 2D menu designs using motor and cognitive models. Given the extensive literature, we direct the readers to [6] for a comprehensive survey.

Hornof et al. proposed EPIC, a cognitive architecture to predict target selection time. Pfeuffer & Li modelled the performance of mobile grid menus based on the number of items and user behavior [60]. Cockburn et al. proposed that the performance of completing an action in a menu could be separated into *item pointing time* and *item search time*, where the latter was linearly correlated with the number of candidate selectable items (N) in the menu interface for inexperienced users [20]. Cockburn and Gutwin also found that for menus with random item ordering, the performance is linearly correlated with N. However, for alphabetically-ordered menus, the time is linearly correlated with the $log_2(N)$ [19]. How does the interface perform as the number of items/applications in the interface increases remains an open question in AR/VR. We define this as the ***scalability*** issue, which we consider crucial to AR/VR applications given its ability to overlay large amount of information in front of the users. Our work serves as an initial exploration of the role of *scalability* during transitional activities in AR setting.

3 Interface Designs

To explore the best ways to support on-the-go access of AR content when users move across multiple locations, we distill three design considerations. Although these design considerations have been well-articulated in prior work on AR interface designs, in this work we took a novel perspective by exploring how these guidelines could be further employed and combined to deploy scalable AR interfaces in transition scenarios.

- **DC1. Availability of information [14, 24, 34, 51]**: how many applications are readily available after the user moves to a new space
- **DC2. Clutter caused by the virtual content [7, 55]**: how obtrusive or cluttering the virtual apps are over the physical environment.
- **DC3. Level of required manual interaction [31, 56]**: how much effort is needed from the users to manually open the apps they need.

When the user transition to a new space, an interface with a low availability level minimizes the level of clutter, thus keeping the real world visible. However, it relies heavily on the user's manual input to open new apps. In contrast, an interface with a high availability level reduces such manual effort required to manually open any AR apps, but may come with the cost of cluttering the scene and bringing in less relevant information.

Consider an example scenario in the life of a typical everyday AR user. The user has a number of AR applications opened and placed in one location (e.g., the living room) (see Fig. 2 (a)). They rely on these AR apps to quickly obtain and be aware of certain information to assist their real-life tasks. At some point, the user decides to transition to another location (e.g., the kitchen) for some tasks. We propose three interfaces that reflect different levels in each design consideration to support the user's tasks after spatial transitions:

3.1 None

The first condition, *None*, represents the existing solution to access a piece of AR content in a new space. In the *None* condition, when users arrive at a new location, all virtual content is still in the old location. To retrieve a piece of AR information placed in the previous location,

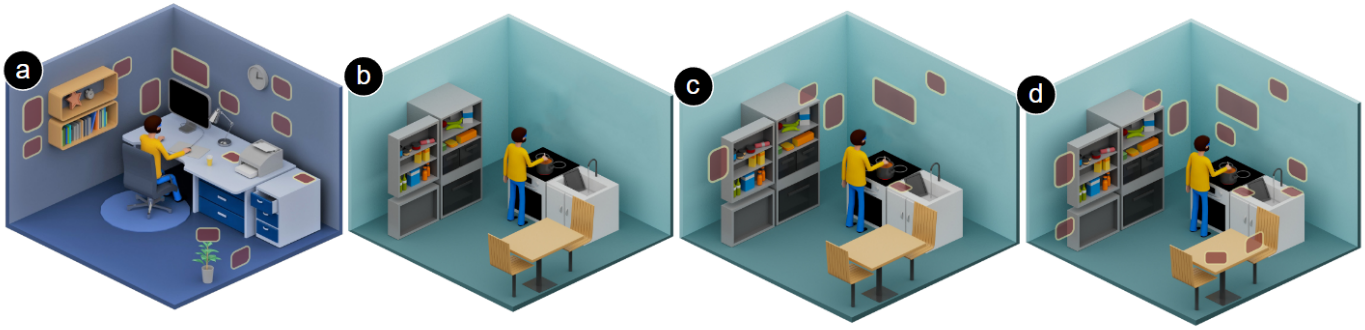


Fig. 2. Example scenarios of using the three conditions: (a) the user is in the office with a set of AR apps opened and placed around, and decides to go to the kitchen. (b) *None*: in the kitchen, the user loses access to the apps in the office, thus having to use a menu to (re-)open the AR apps that he needs. (c) *Some*: the system opens the kitchen-related applications automatically for the user (e.g., recipe, timer, shopping list, fitness). However, the user would still need to use the menu to open other apps. (d) *All*: the system automatically opens all AR applications and places them in the kitchen.

users have to visually search for the app in a menu and reopen it manually (see Fig. 2 (b)). This condition, therefore, has low availability of information and a high level of required manual input, but avoids cluttering the new location with apps that might not be needed.

3.2 All

In the second condition, *All*, the system automatically brings all AR content with the users to the destination, whether it is needed or not. As such, users would be able to access any AR application without reopening it (see Fig. 2 (d)). However, if the number of AR applications increases, it may cause visual clutter and increase the time for the user to find a target application. This condition features high availability of information and a low level of required manual input, but comes with the cost of clutter and intruding on the user's physical environments and tasks.

3.3 Some

As indicated by previous research, location is one of the most frequent trigger of information needs [18, 64]. As such, we believe a system which understands the environment that the user is moving to could be promising and supportive in supporting information needs. In the third condition, *Some*, the system is aware of the new location after the transition process. As such, in the *Some* condition, the system automatically opens the applications relevant to the new location. For example, if users move to the kitchen, the system would open the Recipe, Fitness, and Timer applications automatically and place them at the same relative positions (see Fig. 2 (c)). Therefore, no manual interactions would be needed if users want to access any of the destination-related applications. However, there will also be cases where the user wants to refer to an app that is not opened automatically. If this happens, the user would still need to use the menu to open the application, similar to the *None* condition. This condition features medium availability of information, medium levels of clutter, and a low level of required manual input. We see this condition as a balance between having no and all apps opened, and a promising way of leveraging the spatial capabilities of the AR head-worn devices to seamlessly support the user's transitional activities across multiple physical spaces.

In this work, we thoroughly study the three interface solutions. Through two user studies, we aimed to reach a better understanding of the trade-offs among them both objectively (i.e., performance) and subjectively (i.e., usability, agency, contextual preference).

4 Study 1: Performance trade-off explorations

4.1 Research Questions

In the first study, we aim to conduct an initial and formative exploration of the performance trade-offs of three interfaces in relation to scalability. We aim to answer two research questions in this experiment.

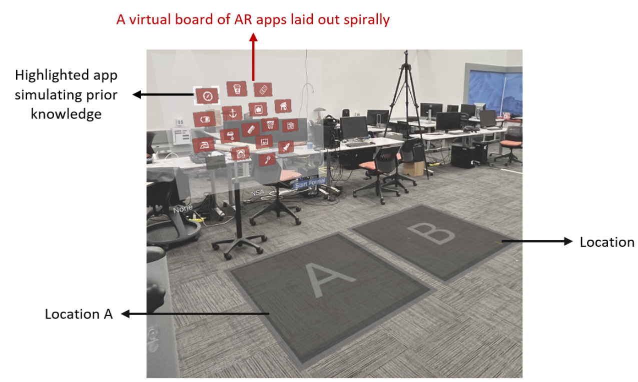


Fig. 3. The task scenario, in which two "locations" were simulated by two squares A and B in AR. Participants were asked to move from one square to another and find a target AR application.

- **RQ1.A.** How does the total number of virtual applications affect user performance of retrieving a target app after a spatial transition?
- **RQ1.B.** What are the trade-offs between the availability level of the AR applications, the required manual interactions, and the visual clutter after the user's spatial transition?

4.2 Study Design

4.2.1 AR apps

We used a Magic Leap One AR headset for the study. Participants had access to a set of AR applications on a virtual board in each of two locations (shown as squares on the floor) in the lab. The virtual board of each square (and the AR apps on the board) was only visible when the user was inside the square. To reduce the potential confounding variables, the AR apps were simulated with a red background and a random icon. They were arranged in a spiral format on the virtual board in front of the users (see Fig. 3). While this setup resembles a 2D menu which has been well-studied in the literature, we believe that the following differences are unique in AR/VR settings: (1) users need to perform physical locomotion in the task with controllers for 3D selections; (2) users need to visually search for targets situated both in the environment and in the menu interface. The total number of AR apps varied from 8 to 28 with a gap of 4 (i.e., 8, 12, 16, 20, 24, 28). We define the number of suggested apps (i.e., n1) in *Some* as 6, which we believe is a reasonable number for users to browse through without causing heavy mental or physical workload. Interaction with the applications were achieved through raycasting, a popular 3D UI technique in AR/VR.

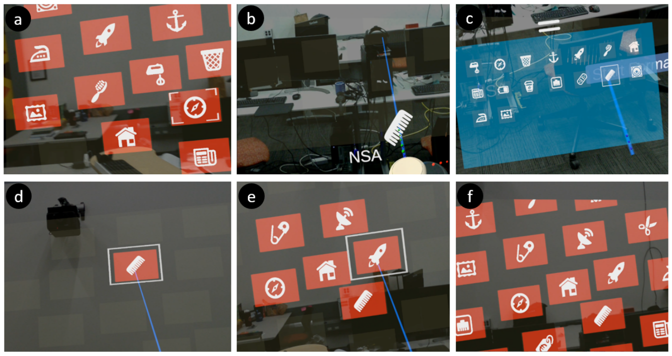


Fig. 4. (a) The system highlighted four apps for users to click through before transition, simulating prior knowledge; (b) the system asked the user to find one of the four apps after transition with an icon appearing on the controller; (c) in the *None* condition, the users pressed a button to use a menu to reopen the app; (d) the user clicked on the correct app using the controller to finish the trial; (e) in the *Some* condition, the system opened the first six apps in the new location, while the rest needed to be accessed through a menu; (f) in the *All* condition, the system opened all apps in the new location. In the tasks, the layout of the apps remained unchanged before and after transition in a single trial, but was randomized between trials.

4.2.2 Transition

By quantifying user performance when the number of AR apps varies, we were able to understand the trade-offs among different interface solutions and study which interface would lead to the optimal performance with different numbers of apps. To simulate the spatial transition procedure (i.e., moving from one space to another) in the real-world as much as we can while collecting sufficient data samples, we designed a task which requires participants to move between two "locations", A and B, back and forth in the real world repetitively. Each "location" was simulated by a 51 inch by 51 inch square on the floor. Six inches separated the two squares (see Fig. 3). In each trial, participants would start in either square A or B, depending on where they left off in the last trial, then transition to the other square and find the target app with the icon indicated on the Magic Leap One controller (see Fig. 4 (b)). At the beginning of each trial when users were located in the first square, the virtual board showed all the AR apps to the users, giving them a sense of the total number of the AR apps and the position of each app. After the user went to the new square i.e., the AR headset enters the square boundary on the floor (see Fig. 3), whether users could still see the AR apps on the virtual board depended on the interface condition. In the *None* condition, all AR apps were invisible (see Fig. 4 (b)), and the user had to click on a button on the handheld controller to activate a menu, point the controller ray at the target icon and press on the trigger button to reopen the target app (see Fig. 4 (c)), then click on it in the virtual board to finish the trial (see Fig. 4 (d)). In the *Some* condition, the system brought six apps with the user and show them on the virtual board at the new location (see Fig. 4 (e)). If the target app was in the six apps, the user could click on it to finish the trial. However, if the target app was not in the six apps, the user needed to open the menu to check the list of unopened apps and open the target app manually, similar to the *None* condition. In the *All* condition, the system brought all the apps with the users to the new location. The user needed to locate the target app and click on it with the controller to finish the trial (see Fig. 4 (f)) .

4.2.3 Simulating prior knowledge with the apps

In everyday scenarios, when mobile phone users want to open an app, they would often roughly know which screen it is and where it is located on the screen due to their familiarity with the layout of the apps. In this study, we tried to simulate such "prior knowledge of the apps" by asking the users to click through four AR apps on the virtual board at the beginning of a trial, and having the target app be one of the four apps after the user transition to the other square. Since the arrangement of the AR app remain unchanged in a trial, the user could locate the target app easier by recalling where it was during the clicking procedure, similar to how we would find an app on our mobile phone.

To summarize, in a single trial, participants were asked to (1) click through four apps in the first location and try to remember what and where they were; (2) walk to the other square; (3) find the target app indicated on the controller on the virtual board; (4) click on it with the controller to finish the trial. In the *Some* condition, the target app was in the six apps that the system automatically opened on the board in half of the trials (we call this condition *Some-Space*), while in the other half of the trials, the user had to open the app manually through the menu (we call this condition *Some-Menu*). As such, participants experienced *Some-Menu* and *Some-Space* conditions all at once as a combined *Some* condition. The trials were randomized so participants had no idea whether the target app would be in the six apps the system suggested prior to each trial. The goal of this was to mimic real-life experiences of using *Some* (i.e., the desired app could be suggested or found manually, which is not known to the users), thus making the derived overall model more valid. The order of the *None, Some, All* interfaces was counterbalanced using a full Latin-Square design We doubled the number of trials in the *Some* condition so the number of trials in each circumstance was equal to the number of trials in either the *None* or the *All* condition. As such, we collected in total 6 (values of N) $\times$ 4 (interface conditions: *None*, *Some-Space*, *Some-Menu*, *All*) $\times$ 3 (repetitions) = 72 trials for each participant.

4.3 Participants & Study Procedure

We recruited thirty participants (9F/21M) from a local university with a mean age of 22 years old ($SD = 3.73$). Fifteen participants had little to no experience with AR prior to the study. The experiment, which was approved by our university ethics board, was divided into six phases. In the first phase, participants were asked to read and sign the consent form. In the second phase, they were asked to fill out a pre-study questionnaire to collect demographic information and prior experience with AR. In the third phase, participants were given a detailed introduction to the experiment background, hardware, the three interface conditions, and the tasks involved in the study. When participants had no further questions, in the fourth phase, we helped participants to put on the AR HWD, and participants were asked to complete the fitting guide program of the Magic Leap One to calibrate the display and determine the ideal size of the forehead-pad and nosepad. Fifth, participants experienced each of the three conditions one by one. The order of the three interfaces was counterbalanced using a full Latin-Square design. Before completing the experimental task in each condition, a training session was provided to get participants familiar with the interactions. After each interface condition, participants were asked to fill out a few questions about what they liked or disliked about the interfaces. Each condition took about three to six minutes. Last, after finishing all conditions, participants were asked to rank the interfaces based on their own preferences and explain their ranking choices in a short interview. The entire experiment took about 30 minutes. Participants were allowed to take a break anytime in between trials.

4.4 Results

We collected in total 30 (number of participants) $\times$ 6 (values of N) $\times$ 4 (interface conditions) $\times$ 3 (repetitions) = 2160 trials. We removed the outliers that deviated by at least two standard deviations away from the average selection time in each interface condition [36, 47]. As such, a total of 88 trials (4.07%) were discarded, yielding a total of 2072 data points. We conducted a series of analyses to explore the trade-offs between the interfaces. For objective measures, we used a two-way repeated-measures ANOVA (RM-ANOVA), with interface condition and N (number of apps) as the two independent variables. A Greenhouse-Geisser correction was applied for violations of sphericity. For subjective measures, we applied Friedman tests and Wilcoxon signed-rank tests for pairwise comparisons. We used an α level of 0.05 in all significance tests. Bonferroni correction was applied to all pairwise comparisons.

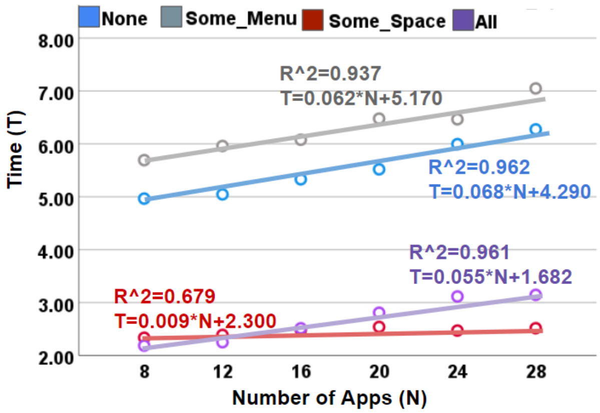


Fig. 5. Regression analysis between the average time (Y-axis) it took for each condition to find the target app and the numbers of apps in the interface (X-axis). Strong linear relationships were indicated for *None*, *Some-Menu*; and *All* conditions..

4.4.1 Time

The results from RM-ANOVA show that both *interface* ($F_{2.083,172.915} = 853.152, p < .001, \eta_p^2 = .911$) and *number of apps* ($F_{5,415} = 45.570, p < .001, \eta_p^2 = .285$) have a significant main effect on the time it took for participants to select the target app after moving to a new position. We also identified a significant interaction between *interface* and *number of apps* ($F_{9.575,794.766} = 3.373, p < .001, \eta_p^2 = .039$). When the number of apps increased, *None*, *All*, and *Some-Menu* led to longer search time, while the time required for *Some-Space* maintained a similar level.

Post-hoc analysis revealed that *None* yielded a significantly shorter time in general to find the target app as compared to *Some-Menu* ($p < .001$), while longer as compared to *All* ($p < .001$) and *Some-Space* ($p < .001$). *Some-Space* yielded a significantly shorter time to obtain the target app as compared to *Some-Menu* ($p < .001$) and *All* ($p = .004$). *Some-Menu* led to significantly longer time as compared to *All* ($p < .001$).

The linear regression was conducted for all interface conditions. As shown in Fig. 5), we identified strong linear relationships (with $R^2 > 0.9$) between the number of apps and *None* ($R^2 = 0.962$), *Some-Menu* ($R^2 = 0.937$), *All* ($R^2 = 0.961$) and a moderate linear relationship (with $R^2 > 0.6$) for *Some-Space* ($R^2 = 0.679$).

4.4.2 User preference

Overall, *All* was selected as the most-preferred interface by 20 participants (66.67%). *None* was ranked as the least-favored interface by 16 participants (53.33%). For *Some*, 8 participants (26.67%) ranked it as the most favored interface, and 14 participants (46.67%) ranked it as the second-favored interface.

4.5 Discussion

4.5.1 Performance trade-offs

There was no intersection between the regression lines of *None* and *All*, which means that in our study, participants would always perform better with *All* as compared to *None* no matter how the number of apps scales. Meanwhile, *All* and *Some-Space* should have a similar performance when N equals the number of suggested apps in *Some*, which was 6 in the study. However, our results show that the two lines intersected when $N = 13.4$. This means that when $6 \leq N \leq 13$, even in the best-cases of *Some* when the users only needed to search within a list of 6 apps, *All* still resulted in a better performance. We surmise that the major reason for this was that the simulated prior knowledge lowered the time it required to find the target apps when N increased in the *All* condition. In the interview, 18 out of the 30 participants (60%) commented that they felt that the procedure of learning where the target app may be located was more helpful in the *All* condition as compared to others. Instead of having to search the entire board, participants only had to search within four "areas", which remained consistent in *All*. Even though remembering the exact locations of the "areas" could become more challenging when the number of apps increased because the areas would often be located outside the FoV, knowing where the target app might be still reduced the required search time by removing the need to aimlessly search in random areas on the board. The fact that all apps were always in front of the users also excluded scenarios where the user has to search around them in order to find a target app, thus leading to a more ideal situation with the *All* interface.

The other factor that may have made the prior knowledge more helpful in *All* than the other two interfaces was the visual consistency of the apps prior to and after transition. One participant commented: *"The best part about this condition (All) is that, since the layout is consistent and static, memorizing the locations of the apps between positions A and B is quite easy."* Such consistency in layout and visibility allowed participants to directly apply the knowledge they learned to the new location, which would not likely be the case in actual real-world scenarios where locations will look quite different. However, in *Some* or *None*, the system only showed some apps or no apps, causing visual distinctions between layouts across the two locations. As such, some additional mental processing was needed to apply the prior knowledge learned. We surmise this helped lower the slope of *All* to be at similar levels with *None* and *Some-Menu* conditions. Without the benefits of such consistency, the additional time required for users to locate a piece of content could be higher with the increment of N, making *All* less applicable when N is large. Our results revealed the benefits of maintaining the physical layout of the AR apps in the space to shorten the time required to find the target app when the number of apps increases.

The design of the menu could also be the reason why participants performed worse in the *None* and *Some-Menu* conditions. The intercepts of the two interfaces which require menu interactions (i.e., *Some-Menu* and *None*) is around three seconds larger than those which does not (i.e., *All* and *Some-Space*) highlight the importance of efficient menu designs. One major issue about the menu was its size. We designed the menu to be small so users could easily skim through it without having to look around too much, which, as mentioned by a few participants, posed challenges for them to pin-point an icon with the controller. The other major issue was about the arrangement of the icons in the menu. Since the apps were simulated by icons without names or semantics, the only way to find a target app in the menu would be to go through the entire list, which would be less likely to happen in everyday situations where the icons would be alphabetically ordered or categorized. With more efficient menu designs, the upper bounds for the derived *None* and *Some-Worst* models could be further lowered.

4.5.2 Understanding the overall performance of *Some*

Our work suggests a linear relationship between selection time T and number of items N, which aligns well with prior work in 2D menu performance modeling by Cockburn and colleagues [19,20]. Assuming such linear relationship holds when N keeps increasing, given the regression result of *Some-Menu* and *Some-Space*, we were able to explore the performance of *Some* overall, which is a weighted combination of *Some-Menu* and *Some-Space* depending on the probability that the needed app is one of the six apps that the system suggested. We adopted two strategies: (1) when the apps are chosen at random, which means a larger value of N would result in a lower probability; and (2) when the probability reflects the performances of state-of-the-art AI systems, such that the likelihood of the needed app being chosen is high, regardless of the value of N. Fig. 6 illustrates the derived performance models for *None*, *Some*, and *All* based on the number of total AR apps.

In the first exploration, we assume that the system selects six apps randomly from the pool of all apps. As such, the probability of the needed app being inside the suggested list would be $\frac{6}{N}$, which decreases with the increment of N. As such, *Some-Overall* outperforms *None* all the time. However, *All* would always be the more efficient option no matter how N scales (see Fig. 6 Some(Worst)).

In the second exploration, we assume the prediction accuracy to be

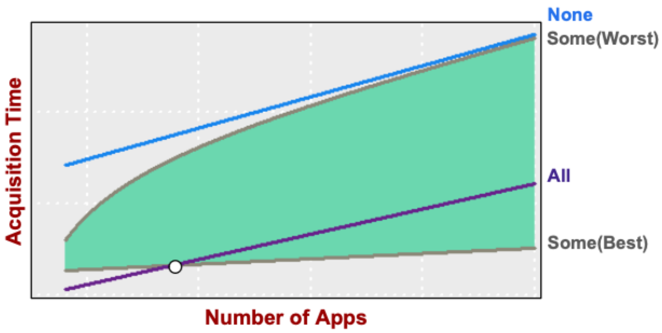


Fig. 6. The final performance models for *None*, *Some*, and *All* with respect to the number of total applications (green area indicates the possible range of performance of *Some*). *None* always performs the worst. The good version of *Some* may outperform *All* when the number of AR apps is large.

at a fixed value. We assume that the accuracy of the needed app being one of the six recommended apps stays at 95%, in line with prior work on recommendation systems [13, 63]. As such, the performance model of *Some-Overall* here symbolizes the best-case overall performance of *Some* (see Fig. 6 Some(Best)). In this case, *Some* would always outperform *None*, and outperform *All* when N is over some threshold.

Overall, through a controlled exploratory study, our results successfully demonstrated that the trade-off exists among the three interfaces when the number of applications differs. It appears that our result strongly favored *All* as the solution for transitioning AR apps on-the-go. However, our study simulates an ideal situation of the *All* condition, in which we assumed (1) all AR apps were directly in front of the user without the need to search around; (2) the layout of the AR apps remained exactly the same before and after the transition; (3) the user was not required to pay attention to the real-world environment that could be occluded by the AR apps. The *Some* condition, although performing worse than we thought, still has merit if the number of apps is larger than a certain number when the system has perfect or nearly-perfect accuracy in predicting the needed app. Therefore, accuracy plays an important role in the usability of the *Some* condition, as a higher accuracy reduced the penalty of having to bring up the menu to retrieve the app manually. One reason for the penalty being high in our study is the design of the menu, as reflected by most participants. The constant needs of calling the menu through a button press and using Raycasting to select small targets made the interaction tedious and time-consuming to perform repetitively. The menu has to be easy-to-use so the users can easily recover when the system does not predict the needed app. Finally, *None*, as a representative of existing solutions, always performed worse than the other two interfaces. However, given the controlled nature of the study, we acknowledge that the models derived here may be limited by its presented task scenario and interface designs. Our results serve as an initial exploration of the possible trade-offs among these interfaces in order to verify our hypothesis and seek ways to further improve our interface designs. We call for more modeling work in the context of AR/VR that explores the *scalability* issues in more realistic setups with content distributed around the user's environment and optimized input.

Our initial study results demonstrated the trade-offs among the interfaces, and that each interface holds characteristics that could make it more usable in a certain situation. However, we need a more ecologically valid study that compare these interfaces in actual AR settings with physical user transitions. As such, we followed up with a second study to further reveal such trade-offs, especially in the situation when the number of needed apps is reasonably large.

5 Study 2: Evaluating Realistic Transitions In-Situ

5.1 Research Questions

In the first study, we revealed the performance trade-offs of each condition and the importance of the menu design for optimized user experience in the *None* and *Some* conditions. In the second study, we aim at exploring the three interface solutions in more ecologically valid scenarios when the number of AR applications is reasonably large. We aim at answering the following research questions:

- **RQ2.A.** With a relatively large number of virtual applications, how well does each interface support information access tasks?
- **RQ2.B.** How do different contextual factors affect the user experience with and preference for the three interfaces?
- **RQ2.C.** What are the trade-offs among the three interface conditions in relatively more ecologically valid simulations of everyday scenarios in AR?

5.2 Interface Conditions

Based on the lessons learned from the first study, we made a few improvements to the menu interface. Instead of using the controller to select the small icons in a world-referenced menu, which was cumbersome and time-consuming, we adopted gaze + hand interactions [61] (i.e., eye-tracking for selection and a hand-referenced menu) to further lower user friction of interactions, making it more applicable to everyday tasks when a separate input device is likely unavailable. Users could simply gaze at an icon in the menu to interact with it. The menu appears next to the hand when the hand is in view (see Fig. 7 (b-d)). To avoid the "Midas touch effect [45]", we adopted an one-second dwell time to strike a balance between speed and accuracy [53, 55]. Inspired by Pfeuffer et al. [59], we designed a two-stage dwell (reveal + selection). As such, participants start by settling their gaze at the same target for 0.5 second to initialize the dwell and reveal the visual indicator (this was to prevent the visual feedback from always appearing when users scan through new menu items). Users then see a white outline around the app icon gradually appearing as visual feedback (see Fig. 7 (c)). Then an additional 0.5 second dwell was required to serve as confirmation, leading a total of one-second dwell time, following prior work on gaze-based activation techniques [53, 55]. After gazing at an icon in the menu for more than one second, the application would be spawned from the menu location and smoothly move to its assigned location in the space. As such, the user could perform quick interactions with the AR apps without needing both hands, which would be convenient in situations when users need to have their hands occupied by tasks in the real world. In addition, we displayed the apps in alphabetical order in the menu to make them easier to find.

We used the same three interface conditions as in study 1:

- *None*: After a user transition from one location to another, users would lose access to all the apps they opened in the previous location. Apps had to be opened manually from the menu, and the icons of opened apps became opaque in the menu.

- *Some*: After a user transition from one location to another (i.e., when the AR headset enters the bounding box of the new space), the system would automatically open the apps that are relevant to the new location. In this study, the bounding box for each space were predefined to encompass all the interactable physical spaces. In real applications, the volume could either be the bounding box of the scanned environment or manually defined by the user. The new apps opened by the system would have their icons become opaque with a blue star on one side to indicate that they had been opened automatically by the system (see Fig. 7 (b-d)). The users could make edits to the system-suggested apps by closing them or opening new ones. To help the users keep track of what apps are already opened and what are not, as commented by participants in study 1, we added visual indications at the top of the FoV. The apps opened in the current space will have their icons displayed there for quick glances (see Fig. 7 (a-e)).

- *All*: Similar as *Some*, after the user's camera enters the pre-defined bouncing box of a new space, the system would automatically open and place all applications. The user could make edits by closing the apps with the menu. However, all apps would still appear automatically at the next location.

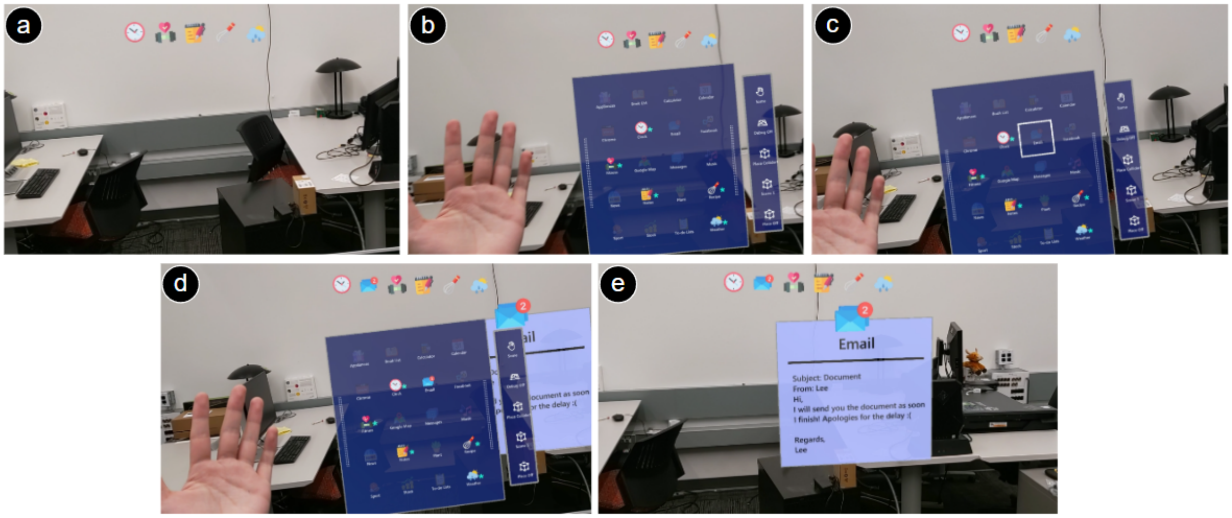


Fig. 7. A demonstration of using the menu in the *Some* interface: (a-b) The user raises the hand to summon the menu, the currently opened/closed apps were shown as opaque/transparent icons, and the system automated apps have a blue asterisk mark on the side; (c) the user gazes at the icon for one second to open an app manually (a white border gradually appears around the icon as visual feedback for the dwell interaction; the icons in the menu are alphabetically ordered for easy search); (d) the icon turns opaque, indicating that the app has been opened in space; (a-e) For *Some*, the icons of the opened apps were shown on the top of the field of view.

5.3 System Implementation

The prototype was implemented on the Microsoft HoloLens 2. A total of twenty AR apps were integrated into the system, including *Appliance, Book List, Calculator, Calendar, Chrome, Clock, Email, Facebook, Fitness, Map, Message, Music, News, Notes, Plant, Recipe, Sports, Stock, To-do List, and Weather*. Each app consisted of an icon, some images and/or a few lines of text. Each app had its own predefined position at each location, which was designed to be consistent across the three spaces. For example, the calendar app would always be located on the right side after the user arrived at a new location facing the entrance orientation. The reasons for this design choice were two-fold: (1) an app would be placed automatically after being opened without the need for the users to manually place it, which has been proven to be important while relying on AR UIs for real-world tasks [56]; (2) the knowledge that users gain about where the app was located in one location applied to all the other locations to maximize learnability and minimize the required mental effort, as shown in the first study. Users were not allowed to change the location of the AR apps, as our study focused on the spatial transition rather than the placement or arrangement of AR content.

5.4 Contextual Framework and Hypotheses

To study the user experience of these interfaces in a deeper and more nuanced way than in study 1, we designed study 2 to examine the influence of three major contextual factors that we hypothesized would impact the user experience:

- ***F1: Real world (RW) Priority.*** The first factor identifies whether occlusion, distraction, or cluttering issues of the AR apps is a problem in the current context, which is defined by how prioritized the real world is in a given context. If the real world is prioritized, which means that the task requires careful monitoring or engagement with objects in the real-world environment, users may only want to open the necessary AR applications to avoid occlusion, distraction, and visual clutter [21,59]. Hence, the *None* may be more favored than *Some* and *All*. If the real world is not prioritized, which means that the current task does not require the user's attention on the real-world environment, then users would be less concerned if more AR apps are present in the physical world. As such, the *Some* or *All* conditions may be ideal.
- ***F2: Workload.*** The second factor relates to how heavy the workload is for the user's current task, indicating the bandwidth users have available to interact with the AR applications. In lightweight scenarios when the primary task is simple, the users would have high cognitive bandwidth available, thus they may not care even if they need to spend more effort to open an app manually. As such, all three interfaces may work well. However, if the user is engaged in heavy tasks in the real-world or the task is time-sensitive, users may have little cognitive bandwidth available. In such situations, they would want to minimize the effort needed to perform manual interactions to open an app, which brings advantages to the *Some* and *All* conditions.
- ***F3: Task Relevancy/Specificity.*** The third factor refers to the availability and relevancy of the AR apps to the tasks at hand, which also means how specific the purpose of the task is. If the user has a primary task with a specific purpose, which means that users have a good idea of what apps they need for the given task at the moment, the availability and relevancy of the AR apps would be at stake to the users because the tasks highly depend on the apps. In such cases, the *Some* interface may be best, since it provides access to the most relevant apps for the current location automatically. However, if the purpose is non-specific (e.g., when the users has no specific task (i.e., just "hanging out") but may want to browse some AR information if they think of it, or where the user has a task that is unrelated to any AR app, but may want to browse some AR information as a secondary task if they think of it), then the *None* and *All* conditions may be best, since they allow the user to open or browse any app as desired.

In the study, participants experienced a set of predefined scenarios which covered different combinations of these factors. Please refer to the Appendix A for more details regarding the definitions.

5.5 Study Design

5.5.1 Scenarios & Tasks

To understand how these contexts impact the user experience of the three interfaces in AR, we designed a user study with three goals: (1) the task needs to simulate everyday scenarios that involve on-the-go information needs with high ecological validity; (2) the task needs to happen in a realistic AR setup in a scenario that the users are familiar with; (3) the task needs to cover the six contexts in a balanced manner.

As such, we designed a study which involves embodied transitions in a large multi-room physical space while wearing an AR HWD.

Table 1. Characteristics of the three physical rooms used in the study and the list of automated applications in the *Some* interface.

Room Space	Physical objects in Room	Automated AR applications
Office	laptop, desk, chair	Email, Calendar, Message, Clock, Weather
Kitchen	fridge, coffee maker, sink, kitchen counter	Recipe, Fitness, Note, Clock, Weather
Living Room	TV, plant, bookshelf, table, chair	News, Stock, Reading List, Plant, Clock, Weather

Participants were asked to go through a set of role-playing scenarios in the real-world environment while using the three interface conditions to obtain information. In the scenarios, participants played the role of an everyday AR user who moves around their own "home environment" for everyday tasks while needing discretionary access to some AR apps on-the-go. The "home environment" was a public indoor area located on our local campus. It consisted of a kitchen, a living room, and a home office (see Table 1 and Fig. 8). In each scenario, the user was asked to start in one of the three locations, then transition three times, each to a different location, and access some AR apps at the new location. To cover the three contextual factors we mentioned earlier, we adopted different combinations of the six contextual situations in each scenario (i.e., see Appendix A - ***C1-C4, C7, C8***). Each scenario included three out of the six contextual situations, each triggered after a transition. As such, a set of four scenarios would lead to twelve contexts, covering each of the contextual situations two times in a balanced form. It is noteworthy that we simulated these contextual factors to the best of our abilities by (1) prompting participants about the current task and setting using voiceovers (see section below); and (2) asking participants to complete the simulated tasks that align with each setting (please see Appendix B for examples of the scenario/task designs). For example, in light workload scenarios, the task happened in a stationary setting with less physical movements, while high workload tasks require users to physically move around and/or visually search for targets.

To motivate the participants to transition to a different location, refer to information in the AR apps, and conduct real-life tasks without breaking immersion of the in-situ role-playing experience, we integrated voiceovers of both third-person and first-person narrations as instructions to guide participants on what room to go to next and what task activity to perform. For example, for first-person narrations, participants would hear *"It has been a long day of work! I'd love to make a coffee in the kitchen;"* or *"Oh no, the guest will arrive within an hour. I need to clean the rooms as fast as possible!"* For third-person narrations, which usually followed right after a first-person narration, participants would hear *"You head over to the kitchen and make a cup of coffee;"* or *"You search for trash in the room and collect it."* The task for the participants was to follow the narrations closely and complete the activities implied in the narration the same as how they would approach them in everyday life. In the narration, participants were prompted to check a piece of virtual content to assist their physical activities, such as *"you grab the ingredients in the fridge based on the Recipe app;"* or *"you check the BookList app to see what books you need from the bookshelf."* Participants then need to check the corresponding virtual app, verbally read the content in the app, and complete the prompted physical activity (e.g., grab ingredients, search for books). The narrations were triggered by the experimenter who walked alongside the participants through a wireless Xbox controller so that the narration smoothly proceeded after participants successfully accomplish the previous step. The order of testing the interface was balanced through a Latin-Square. To mitigate potential learning effects, the order of experience the four scenarios was randomized for each interface. We also randomized the displayed content in the AR apps at the beginning of each new scenario, and asked participants to follow the narration and check the apps each time they were asked to.

5.5.2 Prediction Accuracy in *Some*

Table 1 shows the automatically opened applications for the *Some* interface condition in each location in our prototype. The Clock and Weather applications were automatically opened in each location, due to their versatile uses and frequent needs in everyday cases. Other than that, each space had 3-4 applications relevant to the location and the tasks that happened there, and which would automatically open when the user entered that space with the *Some* interface.

In the four scenarios we designed, we simulated good prediction accuracy for the *Some* interface, backed up by recent successful research in predicting information needs of users. Among the twelve transitions with information queries in our four scenarios, three of them (25%) *Some* did not provide the application required; thus, the users had to manually open the app with the menu, simulating a 75% accuracy level. We wanted to explore how users perceived their experience with the *Some* interface with relatively good accuracy.

5.5.3 Participants & Study Procedure

We recruited twenty-seven participants from a local university. P1 was used as an initial pilot, and P11 & P27 were discarded due to incomplete data logs caused by software issues. The study ended up with 24 participants (4F/20M) with a mean age of 21 years old (SD = 1.13) with complete data. Fourteen participants had little to no experience with AR prior to the study.

The experiment, which was approved by our university ethics board, was divided into six phases. In the first phase, participants were asked to read and sign the consent form. In the second phase, they were asked to fill out a pre-study questionnaire to collect demographic information and prior experience with AR. In the third phase, participants were given a detailed introduction to the experiment background, the concept of AR, the hardware we used, the three interface conditions, and the tasks involved in the study. When participants had no further questions, in the fourth phase, we helped participants to put on the AR HWD, and participants were asked to complete the eye-tracking calibration. Fifth, participants were asked to open the experiment program and were provided a detailed training session to get familiar with the AR applications, their spatial locations, and how to open/close an app using the menu interface. Participants were guided to have an initial with each of the ~~the~~ three interfaces in each room space. They were encouraged to spend extra time in the *Some* condition, in order to learn the locations of the applications that the system would automate. The reasons for this were two-fold. First, it would be challenging for participants to remember the locations of all twenty applications from the the training session. For ecological validity, we wanted participants to hold partial knowledge of the apps' locations, similar to people's knowledge of app locations on a mobile phone. Second, the apps in *Some* have a higher probability of being needed, given that location is often the trigger of information. In the training, participants were told that "Here are some applications that you will likely need later, so it would be helpful if you have some idea where they are located," to avoid biasing them towards a specific interface condition.

After finishing the four scenarios for each condition, participants were asked to fill out a post-study questionnaire with a few questions about what they liked or disliked about the interfaces. Each condition took about 10 to 15 minutes. Last, after finishing all conditions, participants were asked to rank the interfaces based on their own preferences and explain their ranking choices in a semi-structured interview. The entire experiment took about 90 minutes. Participants were allowed to take a break anytime after they finished a scenario, and were encouraged to do so after they finished each interface condition.

5.5.4 Measures

Objective measures. To evaluate user performance on the tasks, we computed: (1) the task completion time, i.e., how long it took a participant to finish all four scenarios using an interface condition; (2) distance travelled: how far the participant walked on average in the study sessions of each interface; (3) amount of head rotation: how much the user rotated their head each second, which could reflect how much effort users spent looking for content in AR and/or the physical

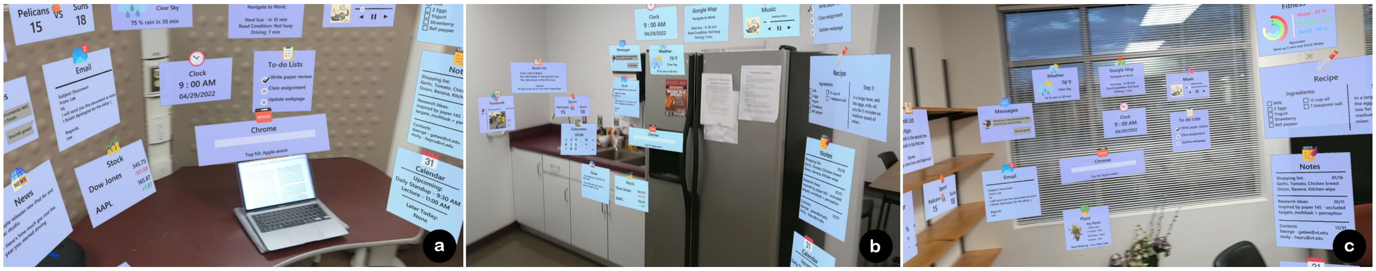


Fig. 8. Deployments of the AR apps (i.e., the *All* interface) in the three physical room spaces for the transition task in study 2: (a) office, (b) kitchen, and (c) living room.

surroundings. We consider these results less relevant due to the less controlled nature of our study. We present these results in the Appendix C.

Quantitative subjective measures. We used the System Usability Scale [9] and two questions from the NASA TLX workload questionnaire [40] to gauge the usability, physical, and mental workload of each interface. We also asked participants to rate their level of agency with each interface using three questions adapted from Tapa et al. [67] to gauge the user's sense of being in control over the AR system. We also measured how accurate they perceived the system automation to be for the *Some* and *All* conditions using a single question "I felt that the automated applications were accurate when I arrived at a new room."

Qualitative subjective comments. In the interview, participants were asked to comment on what they liked and disliked about each interface condition. Additionally, they were asked to dive deeper and think about how each of the contextual factors ***F1: RW Priority***, ***F2: Workload***, ***and F3: Specificity of Needs*** impacted their preference.

5.6 Results

Similar to the first study, we used a one-way RM-ANOVA, with interface condition as the independent variable. A Greenhouse-Geisser correction was applied for violations of sphericity. For subjective measures, we applied Friedman tests and Wilcoxon signed-rank tests for pairwise comparisons. We used an α level of 0.05 in all significance tests, with Bonferroni corrections applied to all pairwise comparisons.

5.6.1 Usability & workload

RM-ANOVA identified a significant main effect of interface on the SUS score with a large effect size ($F_{1.604,36.889} = 10.343, p < .001, \eta_p^2 = .310$). Post-hoc pairwise comparisons showed that *Some* (M = 84.79, SD = 1.74) received a significantly higher usability score as compared to *All* (M = 70.833, SD = 2.40) ($p < .001$). The difference between *None* (M = 80.21, SD = 2.68) and *All* was borderline significant ($p = .066$).

Regarding workload, Friedman test yielded a no significant main effect of interface on the physical ($\chi^2(2) = 7.098, p = .076$) and mental ($\chi^2(2) = 3.528, p = .171$) workload levels.

5.6.2 Perceived agency & accuracy

Fig. 9 (a) shows the average agency rating for each interface. The Friedman test revealed a significant main effect of interface on agency scores ($\chi^2(2) = 7.708, p = .021$). Wilcoxon signed-rank tests found that the *None* condition led to a significantly higher level of agency than *All* ($Z = -2.620, p = .027$). No difference was identified between *None* and *Some* ($Z = -2.201, p = .084$) or *Some* and *All* ($Z = -1.303, p = .576$).

Fig. 9 (b) shows the perceived accuracy level of the system automation in *Some* and *All*. A Wilcoxon signed rank test found that participants considered the automated applications after transitioning to a new space to be more accurate in *Some* than *All* ($Z = -2.049, p = .040$).

5.6.3 Overall user preference

Fig. 9 (c) shows the preference ranking distribution of the three interfaces among the 24 participants. Our results show that the *Some* interface was favored by most participants (66.67%), with only one vote as the least favorite. In contrast, 19 out of 24 (79.17%) participants ranked *All* as the least favored interface, with only one vote as the most favorite. 13 participants (54.16%) ranked *None* as the second favorable interface.

Table 2. The number of votes each interface received under each contextual situation, with the interface condition with the most number of votes bolded.

Contextual Factors	None	Some	All	No Impact
RW Prioritized	**17**	13	0	1
RW Not Prioritized	3	**16**	6	1
High Workload	11	**12**	2	4
Low Workload	6	**12**	4	4
Specific Needs	4	**21**	0	1
Non-specific Needs	**13**	8	2	1
Total Votes	54	**82**	14	12

When participants were asked what they liked about *None*, frequently appearing themes included *easy/intuitive to use (8)*, *being in control (7)*, and *less cluttering or occluding the real world (5)*. However, participants disliked *having to raise hand/reopen applications each time (9)* and *the wait in the dwell (3)*.

For the *Some* interface, participants liked *automatic opening of relevant/necessary apps (12)*, *semantic connections (9)*, *ease of use/helpfulness (7)*, and *not cluttered (3)*. However, it was criticized for *sometimes blocking the real world (4)*.

For *All*, participants liked it because *apps were already accessible (5)*. However, it received criticisms on *being cluttered/overwhelming/distracting (14)*, *covering the real-world environment (12)*, and *opening apps that are irrelevant to tasks/space (5)*.

5.6.4 Contextual preferences

Other than the general preferences, in the semi-structured interview, participants were asked to dive deeper and think about how each of the contextual factors (**F1**, **F2**, and **F3**) impacted their preference. We organized participants' votes on the three interfaces when the three contextual factors change (see Table 2).

F1: RW Priority. When asked whether having to pay attention to the real-world or not in the task had an impact on the interface preference, 23 out of 24 (95.83%) participants gave a positive response.

In the *RW-Prioritized* situations, which in the scenarios required searching for or collecting objects placed in the physical world, *None* received 17 votes (70.83%), followed by 13 votes (54.16%) for *Some*. The *All* condition did not receive any votes. Participants favored *None* because it offered *clear real-world visibility*. Comments included: *"The fact that they were out of view unless I opened them, made it so much easier to find objects or pick up trashes (P2);"* and *"If I have to do tasks in the real world, I would prefer to have minimal applications open (P23)."* The common themes for not choosing *All* included *visibility*, *occlusion*, and *clutter*. For example, participants commented: *"For All it was annoying because I need to look for stuff in the RW but it was all taken up by the AR apps (P15);"* For *Some*, participants who voted for it did not find it too intrusive to the task and think it is acceptable to use: *"I did not find it that obstructive (P7);" "I feel like the Some was*

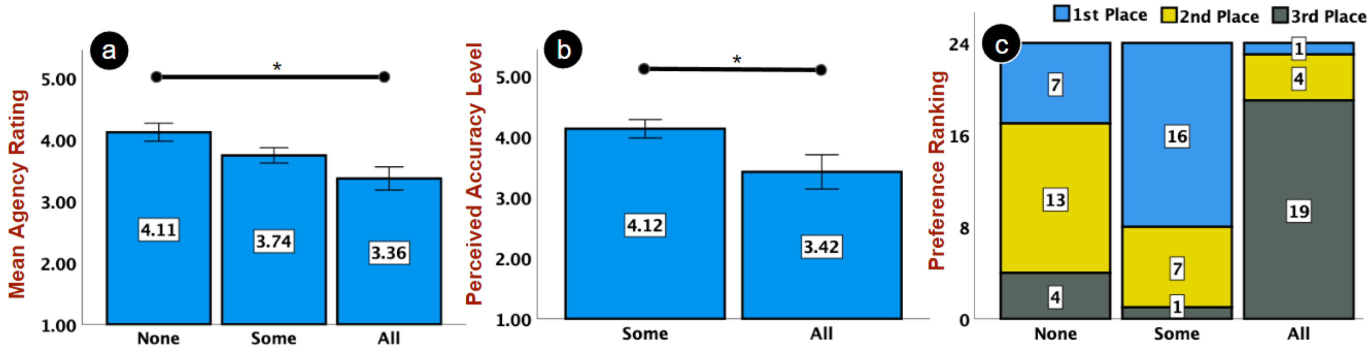


Fig. 9. (a) The average ratings for the three questions regarding Agency; (b) the perceived accuracy level of the automated apps ($\pm S.E.$); and (c) the overall preference rankings of the three interfaces.

good ... it did not block too much of my vision and I could still keep track of the other stuff [information in the apps] I may want to know at the circumstance (P22)."

In the *RW-Not-Prioritized* situation, which in the scenarios were tasks that did not require monitoring/visual searching in the physical world, *Some* received the most votes (66.67%), followed by *All* (25%), and *None* (12.5%). The majority of participants felt that it was beneficial having at least some of apps automatically opened if less attention is needed on the physical world. They commented: *"When I do not need to interact with the real world too much, I would prefer the Some condition more, because it was very convenient (P3);" "I like, for example, in the living room having weather and clock open because it is something I would always use (P20)."* Participants liked *All* because they thought it was *easy, always available*: *"If I wasn't gonna have to interact with the RW a lot, I would choose the All one, because it is easy and everything is there already (P2);"*. However, participants who did not vote for *All* did not share this opinion: *"I think I always need to pay some attention to the real world ... if there is one day that I do not want to leave my room and only get food in the fridge, I would maybe have, not All, but a lot of the apps on that day (P13);"* and *"I would still prefer Some, because All is too much unnecessary apps (P25)"*.

F2: Workload. When asked whether current workload of the task (high vs. low) had an impact on the interface preference, 20 participants gave a positive response (83.33%). The other four participants did not consider workload to have an impact on their preferred interface.

In *high-workload* situations, which in the scenarios required participants to walk around, look on the bookshelf, and grab objects within a time limit, *Some* received the most votes (50%), followed by None (45.83%), and All (8.33%). Participants who voted *Some* favored its *system automation*: *"In heavy duty task, I wouldn't have the mind to open the app by myself (P4);" "I am running around, it does save that time to have Some. I would prefer to have the app already there (P3);" "When I have a lot more going on ... spending the time on opening the apps would not make sense (P16)."* Participants who voted for *None* felt that it was less distracting and does not get in the way in demanding situations: *"In more heavy scenarios ... I would not want the distractions from the apps to my activity (P18);" "If there is window in the way when I am in a hurry, I would get really frustrated ... In real life, without this technology ... if I want to check my email, I would go to my laptop and do that [open apps manually]. So I think I am already getting used to that, it is a traditional experience (P24)"*.

In *low-workload situations*, where participants were relaxing or conducting simple tasks, *Some* received the most votes (50%), followed by None (25%), and All (13.33%). Most participants still did not feel like using *All*, because of its *distractions* and *clutter*. Participants who liked *Some* favored its *convenience* and *suggestions*: *"for lightweight task, I would still use Some for convenience (P13)"; "When I am in a relaxed way with no activity, I would prefer the Some condition because it has a lot of options around (P20);" " I could see myself using the All condition if I was just relaxing ... but still too much to look at. I would still stick with Some, check those suggested apps and pull up the stuff I need (P23)."* Participants who voted for *None* reasoned that they think they would have the bandwidth to open the app manually in non-demanding cases: *"in lightweight case, I would be okay using None because I can easily open the app when I need (P19)."*

F3: Specificity of needs. When asked whether the specificity of the task had an impact on interface preference, 23 participants gave a positive response (95.83%). One participant did not consider their preference to be influenced by this factor.

When the purpose was *specific*, where participants needed specific apps such as needing the recipe app to pick out ingredients from the fridge, 21 out of the 24 participants (87.5%) voted for the *Some* interface, 4 (16.67%) voted for *None*, and no participant chose the *All* interface. The keywords for choosing *Some* over *None*/*All* included *relevant* and *natural*. Participants commented: *"In cases that the apps are specific and relevant, like check the fridge with the recipe app, the apps are super useful. I like having them always there (P2);" "I would prefer having the relevant items be brought up instead of having all of them open in the specific use case (P23);"* and *"Some felt like the Siri suggestion on my phone, it suggested apps I am likely gonna need for my task, which is pretty cool when it works well (P24)."* In contrast, *All* was criticized for being *distracting* and *irrelevant*: *"Having non-relevant apps open distracted me (P20)"*; and *"I see the calculator [app] in the kitchen [for All], I was like I would never need that (P22)."*

When the purpose was *non-specific*, in which participants were either randomly browsing content or checking for apps ad-hoc, such as browsing random apps or checking apps when a notification arises, 13 out of the 24 participants (54.17%) voted for the *None* interface, 8 (33.33%) voted for *Some*, and 2 participants (8.33%) chose the *All* interface. Participants who liked *None* preferred its *controllability* and *on-demand* characteristic: *"In situations that I need random stuff, I would like to be in a little bit control (P8);" "when I need random apps, I could use None and open when I need them (P13)."* Two participants voted for *All*: *"If I am just randomly browsing, it does not bother me, I can just have all the apps there (P15)."* However, the majority of participants think *All* is still too *cluttered* and *irrelevant*, similar to previously listed comments. Participants voted for *Some* because of its potential to fulfill unspecific information needs: *"sometimes the apps I need were already opened by the system, which I found useful (P7);" "you can potentially have the apps ready without the risk of being cluttered (P14);"* and *"...if the prediction is really good [in figuring out what I will need], I would use Some (P23)."*

5.6.5 Desired features

In the interview, participants also touched upon some functions they would like to see included in the three interfaces.

Customizing the Some apps. Seven participants (29.17%) mentioned that it would be great if they could be involved in defining what apps will show up when they enter a new room. Participants commented *"The Some interface may be really good for someone who would like to add the things to different locations so it is more personalized for them (P3);"* and *"the best scenario is you preselect what app I want opened* *in Some. It would be better if I can define what apps are opened in each space (P4);"* and *" I love the convenience of the Some, especially if I could personally set what I wanted in what room, would make things really great (P8)."*

None with pinning. Four participants (16.67%) mentioned that it would be cool to add a *Pin* function to the *None* interface. When they leave the current space, this feature would allow pinned apps to remain open in the new room: *"When I need to carry the note app from one place to another, I would prefer the list to follow me around all the time, like if I can pin the app, so the app just follow me around wherever I go (P24);"* and *"what if the None has a default follow feature, like when I open this app, I want this app to be opened when I move around, that could be helpful so I do not have to reopen the stuff [apps] I used before (P16)."*

Dimming all windows. Two participants (8.33%) mentioned that they would like a function to completely wipe out the AR applications temporarily if the current task requires heavy attention on the physical world. *"If there is a toggle button, where if I wanted, I could use a toggle a switch to make everything disappear just temporarily. When I turn it back, all apps comes back (P24);"* and *"I think it would be helpful to have a gesture to wipe things away temporarily, and bring things back later. When I want to search for something, I could easily wipe things away real quick, to find the site, and bring the apps back. With this, the Some display would be even better (P10)."*

5.7 Discussion & Design Implications

5.7.1 Impacts of the contextual factors

Looking back at our hypothesized trade-offs, our hypothesis about *None* and *Some* being better options than *All* when the *real world is prioritized* was supported by our study. Participants favored the clear visibility of the physical environment that *None* offers. *Some* was also favored by over half of the participants, who did not consider it distracting even when attention to the physical surroundings was required. In contrast, no participants voted for *All*, criticizing its clutter and occlusion. When the *real-world is not prioritized*, our hypothesis partially aligns with the findings, since *Some* was preferred over *None*.

When it comes to *workload*, our hypothesis was partially supported: *Some* was the most favored interface in *high-workload* situations. We hypothesized that *All* would also be favorable since users could avoid the extra effort to open apps manually. To our surprise, participants preferred *None* rather than *All*. The potential distractions of *All* and the need to manually close the apps when they got in the way outweighed the benefits of having the apps automatically opened in demanding situations. In *None*, participants liked the one-handed gaze-triggered menu design, thought it got in the way less, and that it felt similar to how they access information with phones or laptops. This demonstrates the importance of optimizing the manual approach; our redesigned menu made *None* more usable than it was in study 1. In *low-workload* situations, participants favored *Some*, which contradicted our hypothesis about all three interfaces being equally usable. The suggested apps provided by *Some* provided participants with the convenience of leveraging already-available options, while not being overwhelmed by too much information.

Finally, when it comes to *specific vs. non-specific needs*, our results supported our hypothesis by showing that *Some* was the most favored interface in situations when the users need *specific* applications to assist their tasks. As reflected in the ratings, the automation in *Some* was perceived to be more accurate than *All*. Participants considered it natural to have task-relevant applications automatically opened. *All* would have too many irrelevant applications opened in the space, which was considered distracting and unhelpful, also demonstrated by its lower perceived accuracy level than *Some*. When it comes to *non-specific* uses, our hypothesis of *None* being the preferred interface was supported, but not *All*. *None* was favored for its controllability and on-demand nature, which makes it viable for handling the ad-hoc information needs of the users.

5.7.2 *Some* as the sweet spot between *None* and *All*

Our findings in study 2 strongly indicate the value of *Some* in supporting everyday users to access information after spatial transitions. It was the most preferred interface, received higher usability ratings and had a similar agency rating as the manual condition. In general, participants' comments suggested that *Some* is a good balance between *None* and *All*, and nine participants (37.5%) stated this explicitly. It offered more convenient access to relevant applications as compared to the fully-manual approach of *None*, while having less risk of distraction, occlusion, and showing irrelevant applications to the task and spaces as compared to *All*.

Additionally, participants were positive about the semantic connections between the automated applications and the physical objects in *Some*. Participants commented *"Having the recipe app automatically opened near my fridge, plant app opened on my plant, and the email app opened in my office make lots of sense (P25);" "A lot of people have cook books, recipes on their refrigerator, so it make sense to have the [recipe] application there ... it makes more sense to attach the app to the object (P3);"*. Having the AR apps appearing right next to the physical referent to assist the interaction felt natural and useful. Participants also mentioned that such connections could work as a reminder: *"The plant app is nice to have open near my plant, because it is easy to forget that the plant needs watering (P26)."*

In contrast, participants criticized it when the AR apps appeared in ways that did not show connections to the physical space. For example, P10 commented on the *All* condition: *"The plant app is opened in the Office, but the plant is not actually there, which I felt did not make much sense (P10)."* Interestingly, participants also mentioned how the "feel" of the physical space needs to be considered: *"In the kitchen, you are already up and ... walking around. So it makes sense to raise your hand [for the menu], like the none or the some interface. Like in the office, you would want to be more ... cognitive. In the living room, when you are kinda lazy, I would prefer to have the apps already opened for you, like the all and some (P14);"* and *"Having message and mails up in the office makes much sense. But having them in the living room, which is supposed to be a relaxing space, it is not supposed to have email there, you know, work-related stuff (P8)."* This is related to the notion of relevancy. The apps need to be automated or provided in ways that respect the users and their environment, taking into consideration the potential types of activity being done in the particular space. This was well-achieved in the *Some* interface in study 2, in that only the apps relevant to the space would appear, and the tasks that took place were relevant to the space in the design of the scenarios.

6 Synthesizing Study 1 and Study 2

In study 1, we demonstrated the trade-offs among the three interfaces in terms of performance through a controlled simulation study. In study 2, we explored the trade-offs in terms of user preference in more ecologically valid experiences involving physical activities and transitional tasks. We acknowledge that the differences between setups, input modalities, and tasks in two studies limit direct comparisons of our findings. In this section, we focus more on what caused the shared/distinct insights between two studies to distill overall lessons learned.

Our results in study 1 showed that the number of AR applications affects the performance of the interfaces. Our results showed that *All* was the most efficient interface when the *N* was low, but an ideal version of *Some* with good prediction accuracy would achieve better performance than *All* when *N* becomes higher, while *None* was always the slowest interface. In study 2, we observed that *Some* was the most preferred interface that balances between *None* and *All* conditions.

The *All* interface, as the most preferred interface in study 1, became the least favored interface in study 2. Prior work showed that gaze-based interactions may hinder spatial memory of the users [66], which was the primary interaction method to open/close apps in study 2. This may have caused additional challenges for participants to locate applications in study 2. The comments from our participants demonstrated that our second study successfully showcased the potential challenges in more ecologically valid scenarios in study 2 where apps spread around the users and visibility of real world objects are sometimes important. *All* would automatically bring content that is less relevant to the user's task and space, which was criticized by the users. However, given its high information acquisition efficiency as shown in study 1, if the number of apps is low and information acquisition efficiency is highly prioritized, *All* could still be a feasible option.

The *Some* interface, after receiving enhancements according to the major two concerns raised in study 1 about menu design and limited awareness of app status, became much more usable in study 2 and became the most preferred interface. This demonstrates the necessity of minimizing the error-correction cost in *Some* when the app needed by the user is not predicted by the system, which aligns with Horvitz's vision about mixed-initiative interface designs and recent work about automated interfaces in AR/MR [43, 56]. Improving the menu design could be an efficient strategy to lower such cost and bring down the intercept of the models, as demonstrated in this work.

Assuming good prediction accuracy in *Some*, as simulated in study 2, participants liked *Some* especially when they have specific needs on applications for their tasks. In contrast with participants' reactions to *All*, our results suggest that for future MR interfaces, relevancy of information should be prioritized more than just availability of information. This finding aligns with recent explorations on optimization-based adaptive UIs that also took into consideration the frequency of use of each application in current task settings [14, 51]. Our work further suggests that as the number of apps keeps scaling up, such considerations would become even more prominent. Future research should investigate not only how to place AR/VR apps, but also which apps to show in order to best support the user's current tasks.

The *None* interface was the second-most preferred interface in both studies. It offers good real-world visibility, user agency, and works in a similar way as existing familiar devices, making it easy to learn and a feasible back-up option. Participants liked its on-demand characteristic, especially when the real-world attention is prioritized, which some participants mentioned would almost always be the case for them. This demonstrates the importance of enabling not only easy access to the digital apps, but also to the real world simultaneously. In terms of performance, study 1 showed that *None* may be less performant than either *Some* or *None* regardless of the number of apps, since it always requires manual interaction from the users. One solution was to optimize for such manual effort by more lightweight interactions [60]. Another solution could be, instead of showing completely no apps after users move, future research could also investigate non-intrusive and ubiquitous information displays in AR/MR, which could allow users to access virtual information without real-world visibility being compromised. Recent work has demonstrated the potential of such solutions [39, 55]. However, more investigations are needed to take into consideration the type of the AR application, information presented in the application, and information needed from the users to realize such context-aware information displays.

7 Limitations and Future Work

Our studies have some limitations. First, study 1 was conducted in a controlled simulation of transitional activities and was less ecologically valid. This makes the derived model less applicable in real-world scenarios when the content is distributed in 3D and user familiarity/learning effect frequently play important roles. Future research could further systematically analyze the user performance in a variety of settings leveraging existing models for 3D pointing and visual search, similar to [20]. Second, in study 2, we used a public space rather than the user's private spaces. The user's familiarity with the spaces may play a role in their perceptions of the interfaces. Third, although given time to become familiar with the layout and content of the AR applications in each space, the placements of the apps were not defined by the participants themselves. This could have caused negative impacts on the *Some* and *All* interfaces. Fourth, the scenarios, although composed of easy-to-understand tasks that commonly happen in everyday life, were artificial in study 2. Future studies could explore self-triggered uses of the interfaces in uncontrolled everyday scenarios. Fifth, study 2 was conducted in the context of a home environment, which was quiet and indoors. Future research could study how user preferences change in more crowded, open environments. Sixth, the second study is more realistic as compared to prior work given its actual AR setting. However, evaluations of the proposed interfaces in ecologically-valid scenarios are still missing. We call for future research that further study these directions. Seventh, although the scenarios were designed so that they reflect dynamic changes in the three contextual properties, the exact levels were not gauged or controlled. Future research could explore a more controlled setup. Eighth, in study 1, we applied raycast-based selection and switched to hand-based selection in study 2. We acknowledge that this poses a limitation to generalizability of our results across studies. Last, this research assumed condition to trigger the interface transition was clear by applying a pre-defined boundary. Future work could research further whether such method is robust to trigger interface adaptations.

8 Conclusions

Current AR systems offer limited solutions for users to carry previously placed AR content to new spaces for continuous access. In this research, we explored the design of transition-aware interfaces in head-worn AR to assist users in accessing information when moving around multiple physical locations. We explored three interface options: (1) *None*, in which users rely on a menu to manually open an application while entering a new location, representing existing solutions; (2) *Some*, in which the system automatically suggests a subset of the total applications; and (3) *All*, in which the system automatically opens all applications when the user enters a new space. Utilizing the learnings from a controlled study, we identified the performance trade-offs and improved the design of these interfaces. We then compared these interfaces in a more ecologically-valid simulation of everyday transition scenarios. Our results provide valuable lessons on how to design interfaces that are scalable and transition-aware, and how different contextual factors may impact user preference of using the proposed interfaces.

Acknowledgments

We would like to thank all the participants for their time in participating in the studies. This work has been supported by the Office of Naval Research.

References

[1] 40 fascinating mobile app industry statistics [2022]: The success of mobile apps in the u.s. `https://www.zippia.com/advice/mobile-app-industry-statistics/`. Accessed: 2023-01-10.

[2] Mobile app download statistics & usage statistics (2023). `https://buildfire.com/app-statistics/`. Accessed: 2023-01-10.

[3] A. Ajanki, M. Billinghurst, H. Gamper, T. Järvenpää, M. Kandemir, S. Kaski, M. Koskela, M. Kurimo, J. Laaksonen, K. Puolamäki, et al. An augmented reality interface to contextual information. *Virtual reality*, 15(2):161–173, 2011.

[4] S. Al-Megren. A predictive fingerstroke-level model for smartwatch interaction. *Multimodal Technologies and Interaction*, 2(3):38, 2018.

[5] R. T. Azuma. A survey of augmented reality. *Presence: teleoperators & virtual environments*, 6(4):355–385, 1997. doi: 10.1162/pres.1997.6.4.355

[6] G. Bailly, E. Lecolinet, and L. Nigay. Visual menu techniques. *ACM Comput. Surv.*, 49(4), dec 2016. doi: 10.1145/3002171

[7] B. Bell, S. Feiner, and T. Höllerer. View management for virtual and augmented reality. In *Proceedings of the 14th Annual ACM Symposium on User Interface Software and Technology*, UIST '01, p. 101–110. Association for Computing Machinery, New York, NY, USA, 2001. doi: 10.1145/502348.502363

[8] M. Billinghurst and T. Starner. Wearable devices: new ways to manage information. *Computer*, 32(1):57–64, 1999.

[9] J. Brooke et al. Sus-a quick and dirty usability scale. *Usability evaluation in industry*, 189(194):4–7, 1996.

[10] F. Cabric, E. Dubois, and M. Serrano. A predictive performance model for immersive interactions in mixed reality. In *2021 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pp. 202–210. IEEE, 2021.

[11] A. Caetano and M. Sra. Arfy: A pipeline for adapting 3d scenes to augmented reality. In *Adjunct Proceedings of the 35th Annual ACM Symposium on User Interface Software and Technology*, pp. 1–3, 2022.

[12] S. K. Card, T. P. Moran, and A. Newell. *The psychology of human-computer interaction*. Crc Press, 2018.

[13] C. Chen, C. Fu, X. Hu, X. Zhang, J. Zhou, X. Li, and F. S. Bao. Reinforcement learning for user intent prediction in customer service bots. In *Proceedings of the 42Nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 1265–1268, 2019.

[14] Y. Cheng, Y. Yan, X. Yi, Y. Shi, and D. Lindlbauer. Semanticadapt: Optimization-based adaptation of mixed reality layouts leveraging virtual-physical semantic connections. In *The 34th Annual ACM Symposium on User Interface Software and Technology*, pp. 282–297, 2021.

[15] Y. F. Cheng, C. Gebhardt, and C. Holz. Interactionadapt: Interaction-driven workspace adaptation for situated virtual reality environments. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*, UIST '23. Association for Computing Machinery, New York, NY, USA, 2023. doi: 10.1145/3586183.3606717

[16] W. H. Chun and T. Höllerer. Real-time hand interaction for augmented reality on mobile phones. In *Proceedings of the 2013 international conference on Intelligent user interfaces*, pp. 307–314, 2013.

[17] K. Church, M. Cherubini, and N. Oliver. A large-scale study of daily information needs captured in situ. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 21(2):1–46, 2014. doi: 10.1145/2552193

[18] K. Church and B. Smyth. Understanding the intent behind mobile information needs. In *Proceedings of the 14th international conference on Intelligent user interfaces*, pp. 247–256, 2009.

[19] A. Cockburn and C. Gutwin. A predictive model of human performance with scrolling and hierarchical lists. *Human–Computer Interaction*, 24(3):273–314, 2009. doi: 10.1080/07370020902990402

[20] A. Cockburn, C. Gutwin, and S. Greenberg. A predictive model of menu performance. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 627–636, 2007.

[21] S. Davari, F. Lu, and D. A. Bowman. Occlusion management techniques for everyday glanceable ar interfaces. In *2020 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)*, pp. 324–330. IEEE, 2020.

[22] S. Davari, F. Lu, and D. A. Bowman. Validating the benefits of glanceable and context-aware augmented reality for everyday information access tasks. In *2022 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 436–444. IEEE, 2022.

[23] D. Dearman, M. Kellar, and K. N. Truong. An examination of daily information needs and sharing opportunities. In *Proceedings of the 2008 ACM conference on Computer supported cooperative work*, pp. 679–688, 2008.

[24] S. DiVerdi, T. Hollerer, and R. Schreyer. Level of detail interfaces. In *Third IEEE and ACM International Symposium on Mixed and Augmented Reality*, pp. 300–301, 2004. doi: 10.1109/ISMAR.2004.38

[25] M. D. Dogan, E. J. Gonzalez, A. Colaco, K. Ahuja, R. Du, J. Lee, M. Gonzalez-Franco, and D. Kim. Augmented object intelligence: Making the analog world interactable with xr-objects. *arXiv preprint arXiv:2404.13274*, 2024.

[26] B. Ens, E. Ofek, N. Bruce, and P. Irani. Spatial constancy of surface-embedded layouts across multiple environments. In *Proceedings of the 3rd ACM Symposium on Spatial User Interaction*, pp. 65–68, 2015.

[27] B. M. Ens, R. Finnegan, and P. P. Irani. The personal cockpit: a spatial interface for effective task switching on head-worn displays. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 3171–3180, 2014.

[28] J. a. M. Evangelista Belo, M. N. Lystbæk, A. M. Feit, K. Pfeuffer, P. Kán, A. Oulasvirta, and K. Grønbæk. Auit – the adaptive user interfaces toolkit for designing xr applications. In *Proceedings of the 35th Annual ACM Symposium on User Interface Software and Technology*, UIST '22. Association for Computing Machinery, New York, NY, USA, 2022. doi: 10.1145/3526113.3545651

[29] S. Feiner, B. MacIntyre, M. Haupt, and E. Solomon. Windows on the world: 2d windows for 3d augmented reality. In *Proceedings of the 6th annual ACM symposium on User interface software and technology*, pp. 145–155, 1993.

[30] S. K. Feiner. Augmented reality: A new way of seeing. *Scientific American*, 286(4):48–55, 2002.

[31] L. Findlater and J. McGrenere. A comparison of static, adaptive, and adaptable menus. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 89–96, 2004.

[32] P. M. Fitts. The information capacity of the human motor system in controlling the amplitude of movement. *Journal of experimental psychology*, 47(6):381, 1954.

[33] R. Gal, L. Shapira, E. Ofek, and P. Kohli. Flare: Fast layout for augmented reality applications. In *2014 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pp. 207–212. IEEE, 2014.

[34] C. Gebhardt, B. Hecox, B. van Opheusden, D. Wigdor, J. Hillis, O. Hilliges, and H. Benko. Learning cooperative personalized policies from gaze data. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology*, UIST '19, p. 197–208. Association for Computing Machinery, New York, NY, USA, 2019. doi: 10.1145/3332165.3347933

[35] R. Grasset, T. Langlotz, D. Kalkofen, M. Tatzgern, and D. Schmalstieg. Image-driven view management for augmented reality browsers. In *2012 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pp. 177–186. IEEE, 2012.

[36] T. Grossman and R. Balakrishnan. A probabilistic approach to modeling two-dimensional pointing. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 12(3):435–459, 2005.

[37] J. Grubert, M. Heinisch, A. Quigley, and D. Schmalstieg. Multifi: Multi fidelity interaction with displays on and around the body. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, pp. 3933–3942, 2015.

[38] J. Grubert, T. Langlotz, S. Zollmann, and H. Regenbrecht. Towards pervasive augmented reality: Context-awareness in augmented reality. *IEEE transactions on visualization and computer graphics*, 23(6):1706–1724, 2016. doi: 10.1109/TVCG.2016.2543720

[39] V. Y. Han, H. Cho, K. Maeda, A. Ion, and D. Lindlbauer. Blendmr: A computational method to create ambient mixed reality interfaces. *Proc. ACM Hum.-Comput. Interact.*, 7(ISS), nov 2023. doi: 10.1145/3626472

[40] S. G. Hart. Nasa-task load index (nasa-tlx); 20 years later. In *Proceedings of the human factors and ergonomics society annual meeting*, vol. 50, pp. 904–908. Sage publications Sage CA: Los Angeles, CA, 2006.

[41] A. M. Hinze, C. Chang, and D. M. Nichols. Contextual queries express mobile information needs. In *Proceedings of the 12th international conference on Human computer interaction with mobile devices and services*, pp. 327–336, 2010.

[42] P. Holleis, F. Otto, H. Hussmann, and A. Schmidt. Keystroke-level model for advanced mobile phone interaction. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 1505–1514, 2007.

[43] E. Horvitz. Principles of mixed-initiative user interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '99, p. 159–166. Association for Computing Machinery, New York, NY, USA, 1999. doi: 10.1145/302979.303030

[44] S. Houben, P. Tell, and J. E. Bardram. Activityspace: Managing device ecologies in an activity-centric configuration space. In *Proceedings of the Ninth ACM International Conference on Interactive Tabletops and Surfaces*, pp. 119–128, 2014.

[45] R. J. Jacob. Eye movement-based human-computer interaction techniques: Toward non-command interfaces. *Advances in human-computer interaction*, 4:151–190, 1993.

[46] S. Julier, M. A. Livingston, I. Swan, B. Y. JE, and D. Brown. Adaptive user interfaces in augmented reality. In *Workshop on software technology for augmented reality systems (STARS)*. Citeseer, 2003.

[47] R. Kopper, D. A. Bowman, M. G. Silva, and R. P. McMahan. A human motor behavior model for distal pointing tasks. *International journal of human-computer studies*, 68(10):603–615, 2010.

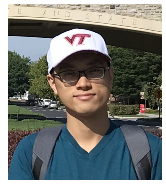
[48] W. S. Lages and D. A. Bowman. Walking with adaptive augmented reality workspaces: design and usage patterns. In *Proceedings of the 24th International Conference on Intelligent User Interfaces*, pp. 356–366, 2019. doi: 10.1145/3301275.3302278

[49] J. N. Li, Y. Xu, T. Grossman, S. Santosa, and M. Li. Omniactions: Predicting digital actions in response to real-world multimodal sensory inputs with llms. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*, CHI '24. Association for Computing Machinery, New York, NY, USA, 2024. doi: 10.1145/3613904.3642068

[50] Y. Li, Y. Hu, Z. Wang, and X. Shen. Evaluating the object-centered user interface in head-worn mixed reality environment. In *2022 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pp. 414–421. IEEE, 2022.

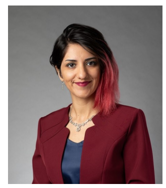
[51] D. Lindlbauer, A. M. Feit, and O. Hilliges. Context-aware online adapta-

tion of mixed reality interfaces. In *Proceedings of the 32nd annual ACM symposium on user interface software and technology*, pp. 147–160, 2019.
[52] D. Lindlbauer and A. D. Wilson. Remixed reality: Manipulating space and time in augmented reality. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, pp. 1–13, 2018.
[53] F. Lu and D. A. Bowman. Evaluating the potential of glanceable ar interfaces for authentic everyday uses. In *2021 IEEE Virtual Reality and 3D User Interfaces (VR)*, pp. 768–777, 2021. doi: 10.1109/VR50410.2021.00104
[54] F. Lu, S. Davari, and D. Bowman. Exploration of techniques for rapid activation of glanceable information in head-worn augmented reality. In *Symposium on Spatial User Interaction*, SUI '21. Association for Computing Machinery, New York, NY, USA, 2021. doi: 10.1145/3485279.3485286
[55] F. Lu, S. Davari, L. Lisle, Y. Li, and D. A. Bowman. Glanceable ar: Evaluating information access methods for head-worn augmented reality. In *2020 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 930–939, 2020. doi: 10.1109/VR46266.2020.00113
[56] F. Lu and Y. Xu. Exploring spatial ui transition mechanisms with head-worn augmented reality. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*, 2022. doi: 10.1145/3491102.3517723
[57] W. Luo, A. Lehmann, H. Widengren, and R. Dachselt. Where should we put it? layout and placement strategies of documents in augmented reality for collaborative sensemaking. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*, CHI '22. Association for Computing Machinery, New York, NY, USA, 2022. doi: 10.1145/3491102.3501946
[58] B. Nuernberger, E. Ofek, H. Benko, and A. D. Wilson. Snaptoreality: Aligning augmented reality to the real world. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, pp. 1233–1244, 2016.
[59] K. Pfeuffer, Y. Abdrabou, A. Esteves, R. Rivu, Y. Abdelrahman, S. Meitner, A. Saadi, and F. Alt. Artention: A design space for gaze-adaptive user interfaces in augmented reality. *Computers & Graphics*, 95:1–12, 2021.
[60] K. Pfeuffer, B. Mayer, D. Mardanbegi, and H. Gellersen. Gaze + pinch interaction in virtual reality. In *Proceedings of the 5th Symposium on Spatial User Interaction*, SUI '17, p. 99–108. Association for Computing Machinery, New York, NY, USA, 2017. doi: 10.1145/3131277.3132180
[61] 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
[62] X. Qian, F. He, X. Hu, T. Wang, A. Ipsita, and K. Ramani. Scalar: Authoring semantically adaptive augmented reality experiences in virtual reality. In *CHI Conference on Human Factors in Computing Systems*, pp. 1–18, 2022.
[63] C. Qu, L. Yang, W. B. Croft, Y. Zhang, J. R. Trippas, and M. Qiu. User intent prediction in information-seeking conversations. In *Proceedings of the 2019 Conference on Human Information Interaction and Retrieval*, pp. 25–33, 2019.
[64] T. Sohn, K. A. Li, W. G. Griswold, and J. D. Hollan. A diary study of mobile information needs. In *Proceedings of the sigchi conference on human factors in computing systems*, pp. 433–442, 2008. doi: 10.1145/1357054.1357125
[65] M. Sra, S. Garrido-Jurado, and P. Maes. Oasis: Procedurally generated social virtual spaces from 3d scanned real spaces. *IEEE transactions on visualization and computer graphics*, 24(12):3174–3187, 2017.
[66] V. Tanriverdi and R. J. K. Jacob. Interacting with eye movements in virtual environments. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '00, p. 265–272. Association for Computing Machinery, New York, NY, USA, 2000. doi: 10.1145/332040.332443
[67] A. Tapal, E. Oren, R. Dar, and B. Eitam. The sense of agency scale: A measure of consciously perceived control over one's mind, body, and the immediate environment. *Frontiers in psychology*, 8:1552, 2017.
[68] M. Tatzgern, V. Orso, D. Kalkofen, G. Jacucci, L. Gamberini, and D. Schmalstieg. Adaptive information density for augmented reality displays. In *2016 IEEE Virtual Reality (VR)*, pp. 83–92, 2016. doi: 10.1109/VR.2016.7504691

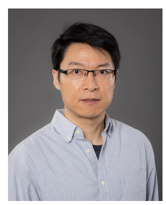


Feiyu Lu obtained his Ph.D. from Virginia Tech in May 2023. His research interests lie broadly in the intersections of AR/VR, 3DUI, and HCI. His Ph.D. work focuses on enabling lightweight and unobtrusive information display and interactions on AR HWDs to support a variety of everyday tasks.

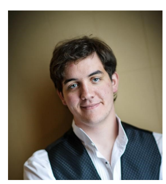
Leonardo Pavanatto obtained his Ph.D. from Virginia Tech. His expertise area is VR/AR within the scope of 3D User Interfaces. Some specific research interests include how to use augmented reality to improve productivity on real-world tasks.



Shakiba Davari is a recent Ph.D. graduate from Virginia Tech's 3D Interaction Group, specializing in the design and development of context-intelligent XR (iXR) interfaces through both UI/UX research and methodological guidelines.



Lei Zhang is an Assistant Professor of game design and development at Kennesaw State University. He received his Ph.D. in Human-Centered Design from Virginia Tech. His research focuses on Human-Computer Interaction, immersive storytelling, serious games, and their applications in education, healthcare, and mental health therapies.



Lee Lisle holds a Ph.D. degree in Computer Science at Virginia Tech. His Ph.D. work centers around Human-Computer Interaction and immersive analytics with multimedia document sets. For his Ph.D. work, he designed and implemented tools in AR/VR to assist sensemaking tasks with non-quantitative datasets.

Doug A. Bowman is the Frank J. Maher Professor of Computer Science and Director of the Center for Human-Computer Interaction at Virginia Tech. He is the principal investigator of the 3D Interaction Group, focusing on the topics of three-dimensional user interfaces, VR/AR user experience, and the benefits of immersion in virtual environments.

A Conceptual Framework for Study 2

Based on the three factors, we defined a framework for categorizing the user's context after transitioning to a new location, leading to eight *contextual situations*,to help us evaluate the impact of these factors on the interface preference in a more balanced and ecologically-valid form.

C1: *RW Not Prioritized + Light + Non-Specific*: The user randomly browses non-specific AR apps without needing to pay attention to the RW environment (for example, casually browsing some random AR apps to kill time).

C2: *RW Not Prioritized + Light + Specific*: The user needs access to some specific apps without needing to pay attention to the RW environment (for example, the user wants to know the weather for today or the number of steps walked in a relaxed setting).

C3: *RW Prioritized + Light + Non-Specific*: The user focuses on a simple task in the RW and sometimes wants to know non-specific or irrelevant information from AR apps. Examples include casually walking or waiting for food in the microwave while browsing some random AR apps.

C4: *RW Prioritized + Light + Specific*: The user focuses on a simple task in the RW while needing access to specific and relevant AR apps to assist the task. Examples include grabbing the ingredients in the fridge according to a virtual recipe or walking on the street according to a virtual map.

C5: *RW Not Prioritized + Heavy + Non-Specific*: The user engages with a heavy digital task while wanting to access non-specific or irrelevant AR applications at the same time. An example would be playing video games with a virtual monitor while monitoring the scoreboard of a football match in the sports app.

C6: *RW Not Prioritized + Heavy + Specific*: The user engages with a heavy digital task while wanting to know specific and relevant AR applications at the same time. Examples include performing productivity work in front of a virtual monitor while needing information from the notes or the browser app to assist the work.

C7: *RW Prioritized +Heavy + Non-Specific*: The user engages in a heavy task which requires continuous attention to the RW while needing access to non-specific or irrelevant AR content. Examples include cleaning the house while browsing the news or monitoring the stock prices in AR.

C8: *RW Prioritized + Heavy + Specific*: The user engages in a heavy task which requires continuous attention to the RW while needing access to specific and relevant AR content to assist the task. Examples include cooking according to a virtual recipe, or assembling furniture according to digital instructions.

We excluded ***C5*** and ***C6*** from our exploration, given that they focus on primarily the interactions with digital tasks, for example doing productivity work with virtual monitors or playing immersive games, with neither the need nor the cognitive bandwidth available for real-world interactions, which happens less frequently during transitional activities. As such, in this work, we focus on ***C1, C2, C3, C4, C7,*** *and* ***C8*** to explore how the three contextual factors **F1**, **F2**, and **F3** could affect user experience and preference of using the three interface conditions under these contextual scenarios.

B Example scenarios used in Study 2

Scenario 1 (*C4 - C1 - C2*):

Starting - Office: You are working in your home office, with *email*, *calendar*, *message* app opened around your desk.

"Ah it's been a long day of work! It's time to make a coffee in the kitchen. I'll also have to pay attention to my email and message apps, just in case my colleague sends me the documents I need."

T1: **Transition 1 - Kitchen:** You go to the kitchen and starts making a coffee while monitoring your *email* and *message* apps.

<Wait for participant to transition to the kitchen>

Suddenly, you saw the fridge in the kitchen environment

"Oh right Let's see what I need to pick up at the grocery store later for the recipe I bookmarked earlier in the *recipe* app"

You open the fridge with the *recipe* app and shopping list in your *note* app, and cross-check what ingredients you'll need to pick up later in the grocery store.

<Wait for participant to check the *recipe*, *note* apps and the fridge>

"Done! Let's go to the living room and take some rest."

[◇ ***C4*:** RW Prioritized + Specific + Light]

T2: **Transition 2 - Living room** You head over to the living room and sit down at the couch.

<Wait for participant to transition to the living room>

"Now let's browse some random stuff to kill time. I wonder what is the stock price of Apple now.. oh and I have not read the trending news for today..."

You start reading the *stock* and the *news* app.

<Wait for participant to open and check the apps>

"Oh it seems that I get the documents I need! Now time to get back to work."

[◇ ***C1*:** RW Not Prioritized + Non-Specific + Light]

T3: **Transition 3 - Home Office:** You head to the office and work on the documents.

<Wait for participant to transition to the office>

You look outside the window and it is a bit cloudy.

"It seems that it is going to rain soon. Should I go to grocery store later? What is the chance of rain?"

You look at the *weather* app to check the chance of being rainy.

<Wait for participant to check the weather app>

[◇ ***C2*:** RW Not Prioritized + Specific + Light]

End

Scenario 2 (*C8 - C3 - C2*):

Starting - Office: "Have been working for 2 hours already. Time to take some rest."

You open the sports app and news app to read them. The sports app catches your attention. "Oh. its Pelicans vs. The Suns. I wonder how the match will proceed."

You go to the living room while keeping an eye on the sport match.

<Wait for participant to transition to the living room>

T1: **Transition 1 - Living room:** The bookshelf on your left catches your attention

"Oh, I haven't been reading for days. I am sure there are a few books in my reading list that I need to finish. Let me grab these books from the bookshelf."

You start to search for books on the bookshelf according to your book list app.

<Wait for participant to check the book list app and search for the books on the bookshelf>

"Hmm I cannot find my wallet. Is it in the office?" You head over to the home office and search for the wallet.

[◇ ***C8*:** RW Prioritized + Specific + Heavy]

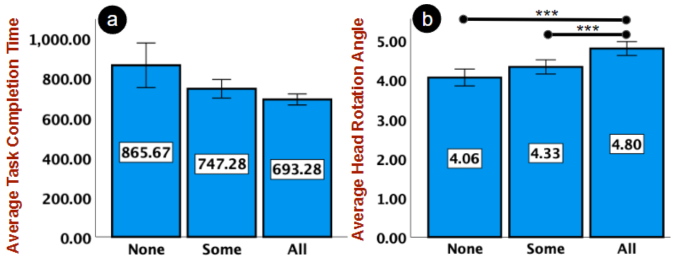


Fig. 10. (a) The average time it took for participants to complete all four scenarios for an interface condition (in seconds); (b) the average head rotation (in degrees/second); and (c) the average System Usability Scale (SUS) score ($\pm S.E.$).

T2: **Transition 2 - Office** You search for the wallet in the office.

[While searching] "I wonder how the basketball match now." You check the sports app.

<Wait for participant to check the sports app and search for the books>

"Not in the office either. Next the Kitchen." You move to the kitchen and search for the wallet."

[$\diamond$ ***C3*:** RW Prioritized + Non-Specific + Light]

T3: **Transition 3 - Kitchen:** You search for the wallet in the office.

<Wait for participant to search for the wallet in the kitchen>

"Ah find it! ... I remember there is a party tonight. When will it be? I need to check my calendar. I should also check the weather and see if I need an umbrella or not." You check the calendar app about when will the party start. You also check the weather to see if it will be rainy around the party time.

<Wait for participant to check the weather and calendar apps>

[$\diamond$ ***C2*:** RW Not Prioritized + Specific + Light]

End

C OBJECTIVE MEASURES IN STUDY 2

C.1 Task completion time and distance travelled

Fig. 10 (a) shows the average time it took for a participant to finish completing the four scenarios in an interface condition. *None* took the longest overall (M = 865.67, SD = 551.42), followed by *Some* (M = 747.28, SD = 228.80) and *All* (M = 693.28, SD = 135.50). RM-ANOVA yielded no significant main effect of interface on the task completion time ($F_{1.202,27.645} = 2.546, p = .117$).

Regarding distance travelled, participants travelled 235.08 units on average for *All* (SD = 4.31), followed by *Some* (M = 231.12, SD = 5.44) and *None* (M = 230.428, SD = 3.63). RM-ANOVA found no significant main effect of interface on the distance travelled ($F_{2,46} = .569, p = .570$).

C.2 Head rotation

We computed the average rate of head rotation for each of the interface conditions (see Fig. 10 (b)). RM-ANOVA yielded a significant main effect of interface on the mean head rotation angle with a large effect size ($F_{2,46} = 18.276, p < .001, \eta_p^2 = .443$). Post-hoc pairwise comparison showed that *All* (M = 4.80, SD = .21) resulted in significantly more head rotation than *Some* (M = 4.33, SD = .18) ($p < .001$) and *None* (M = 4.06, SD = .21) ($p < .001$).