Exploring Spatial UI Transition Mechanisms with Head-Worn Augmented Reality

Feiyu Lu
feiyulu@vt.edu
Center for Human-Computer Interaction
Virginia Tech
Blacksburg, Virginia, USA

Yan Xu
yanx@fb.com
Reality Labs Research
Meta
Redmond, Washington, USA

ABSTRACT

Imagine in the future people comfortably wear augmented reality (AR) displays all day, how do we design interfaces that adapt to the contextual changes as people move around? In current operating systems, the majority of AR content defaults to staying at a fixed location until being manually moved by the users. However, this approach puts the burden of user interface (UI) transition solely on users. In this paper, we first ran a bodystorming design workshop to capture the limitations of existing manual UI transition approaches in spatially diverse tasks. Then we addressed these limitations by designing and evaluating three UI transition mechanisms with different levels of automation and controllability (low-effort manual, semi-automated, fully-automated). Furthermore, we simulated imperfect contextual awareness by introducing prediction errors with different costs to correct them. Our results provide valuable lessons about the trade-offs between UI automation levels, controllability, user agency, and the impact of prediction errors.

CCS CONCEPTS

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

KEYWORDS

agency, automation, controllability, adaptive interfaces

1 INTRODUCTION

Compared to Virtual Reality (VR) headsets where users are immersed in a virtual environment, Augmented Reality (AR) glasses enable people to interact with their everyday physical world with the digital augmentation [3]. In a typical everyday-life activity, people will need to move around to carry out different tasks, changing their information needs on-the-go. Recent research has shed light on the potential of AR glasses to support such needs in common

everyday scenarios [29, 35, 41]. For example, recent work by Lu & Bowman suggested that AR head-worn displays (HWDs) could support easier and less distracting everyday information acquisitions as compared to mobile phones [41]. However, in existing state-of-the-art AR operating systems (OS) (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. This kind of mechanism assumes that the main use cases for AR are confined in one space, limiting the mobility and accessibility of the digital content when users move around.

With mobile computing (e.g., smartphones, smartwatches), people can access different applications and information on-the-go. However, most of the time, these systems still rely very much on the users' effort to find and open the application that is needed at that time. This poses challenges to the users who need to focus on real-world tasks with their attention and hands occupied. How could we enable easier access to the digital content as users move across different environments while needing access to some information?

One direction is to predict what the user tries to do and surface the corresponding functions. With the advancements in Artificial Intelligence (AI) and computational power, recent user interfaces have become more capable of predicting user intent and suggesting potential interactions to be performed by the users [34, 64]. Recently, more and more of this kind of prediction is applied to mobile systems, where the mobile applications make interaction suggestions based on the time of the day, the history of interactions, and location [8, 55, 68, 72]. For example, Google Maps occasionally pops up a suggestion to navigate to a certain destination based on past uses.

We see the great opportunity to leverage prediction and automation with AR systems. AR devices have the potential to understand users' intent accurately and just-in-time, due to the wearability, world-facing sensors (e.g. egocentric videos, depth cameras), and user-facing sensors (e.g. eye-tracking cameras). Combined with the increased AI capability, AR can help offload the users' effort of finding the digital content to the system.

To explore the intersection between AI and AR, a lot of questions need to be answered. First, how would people respond to user interfaces (UI) that try to predict and adapt to their needs? What do they like or dislike about it? Second, how is such automated experience compared to manually controlling the UIs, the latter of which is more familiar to users? Third, how would efficiency, usability, and agency be affected when the interfaces automatically adapt to user needs with different levels of user control? Finally, given that it is virtually impossible for any prediction to reach 100%

accuracy, how would the user experience be affected when the system predicts user intent incorrectly, and how to mediate the consequences when an error happens?

In this research, we answer these questions by designing, developing, and evaluating several mechanisms to spatially transit AR UIs when people move in space. To inform our design directions, we first conducted a body-storming design workshop with expert user experience (UX) designers, in which we learned about the major problems participants encountered when they use AR glasses for acquiring information on-the-go. We then designed and implemented three UI transition mechanisms as outcomes of the workshop. These interfaces have different levels of automation and controllability, which required different levels of user effort to access AR content on-the-go. Moreover, we simulated the inaccuracy/error of prediction about what UI widgets users may need at different locations. With this simulation of errors, we looked into how users perceive and handle the error in the context of spatial tasks while given different levels of user control over the automation results. Finally, we ran a within-subject study with 40 users to compare the three UI transition mechanisms, plus the baseline of the manual UI manipulations which is available on existing commercial AR glasses.

Through the design workshop and the user study, We learned valuable lessons about users' needs for on-the-go AR UIs. We also learned users' performance and preferences with different UI transition mechanisms, which reveals the relation among automation, controllability, and user agency. Furthermore, we found that prediction errors were perceived differently with different controllability and different error-recovery cost.

The main contributions of this work include: (1) explorations of the challenges users encounter when trying to access AR content on-the-go; (2) designing and implementing three interface solutions with different levels of automation controllability; (3) empirical findings about users' performance and preference among the different interfaces and how prediction error affects the experience. (4) design implications for future implementations of automated UI transition mechanisms for AR.

2 LITERATURE REVIEW

2.1 Everyday Information Acquisition with AR HWDs

People encounter a variety of information needs in their everyday lives [14, 21]. AR HWDs have the potential to address such needs by displaying relevant information directly in the real-world environment in front of the users. ARWin is an early attempt of displaying everyday information such as calendar, weather, and clock in AR on a tabletop [22]. In recent work, Colley et al. explored displaying virtual information on top of relevant objects at home to augment user memory [18]. Ventä-Olkkonen et al. explored displaying everyday information on home windows [61]. Knierim et al explored the use of AR for displaying information in home environments [35]. Lu et al. explored displaying everyday information as glanceable UIs at the periphery of the user's view [42]. These work shed light on the potential of AR displays to fulfill people's everyday information needs. Most of these work involved the idea of "widgets", which are compact glanceable UIs for quick access to

information. Widget UIs have been the common form of displaying information on current mobile phone interfaces [62, 67]. Similarly, in this research, we focused on everyday information access in AR systems with widget UIs.

2.2 Mobility of UIs in AR

UIs in AR are usually rendered at a fixed location in the real world. However, in everyday situations. information could be needed on-the-go in a less-controlled manner [15, 57]. As such, recent research has explored the possibility of carrying AR content with the users while moving. Lages & Bowman explored an adaptive walking interface in which AR windows become body-referenced and follow the users around [37]. Lu et al. explored display-referenced and body-referenced layouts for carrying the AR content with the users [42]. The major limitation of these approaches is scalability. Because the system has no knowledge of what the users might need, it has to bring all the AR content that the user will possibly need, while increased pieces of information could cause information overload and distract the users. An early study by Sohn et al. found that 72% of the information needs were prompted by contextual factors such as location changes and activities to be done [57]. In this research, we explored the possibility of automated UI placements based on location changes and activities, and compared them with existing solutions such as display-referenced follow behaviors and manual drag and drops of the AR UIs.

2.3 Levels of System Automation and User Control

Roy et al. defined *automation* as the programming of complex tasks to be automatically executed by a machine with the goal of reducing tedious manual effort, workload and improving productivity of everyday human users [51]. Automated systems may encompass a wide range of low-to-high automation levels and user controls. A higher level of system automation could lead to a lower level of user control because the system would take over, performing more decision-making and task-executions with less user interference. In 2004, Findlater and McGrenere proposed three levels of automation: (1) *adaptive*: the system controls all the interface changes with no user control; (2) *adaptable*: the users control all the interface changes with no system control; and (3)*mixed-initiative*: the control is shared between the user and the system [23]. A more widely adopted automation level standard could be found in the field of autonomous driving. The standard J3016_202104 (SAE 2021) defined six levels of automation in automated driving: *no driving automation (level 0)*; *driver assistance (level 1)*; *partial driving automation (level 2)*; *conditional driving automation (level 3)*; *high driving automation (level 4)*; and *full driving automation (level 5)* [52]. A lower level of system automation allows customization for certain needs, while a higher level of automation reduces complexity and friction to interact with the UIs [71]. In our work, we are interested in understanding how different levels of automation impact users' efficiency and agency.

Automation is powered by the advancement in Machine Learning and AI. The algorithm generates an output ("prediction") based on the past training data, the input, and the model. Some autonomous

systems may be capable of predicting user intent and making decisions on the users' behalf [19, 20]. However, the predictions may be inaccurate. In addition to trying to improve the accuracy of prediction, it's proven to be critical to design for *controllability*, which "reflects to what extent the users can control the automation or alter its result to reach their goal, and how easily and rapidly can this control be carried out [51]." There have been long-time debates on how much the system should be involved in the automation of UI components, as well as how much controllability should the user hold over the automation. Findlater and McGrenere found that *adaptable* was more preferred and significantly more efficient than *adaptive* in 2D menus [23]. Gajos et al. found that *adaptive* interfaces were not necessarily advantageous purely because of their theoretical benefits [26]. Zhang et al. found that combining *adaptive* with *adaptable* could lead to higher usability [71]. Roy et al. found that manual approaches were more preferred as compared to automated systems [51]. However, little research has been explored for AR displays about how automation and controllability would impact the multi-faceted user experience. In this research, we aim to explore the trade-off between automation and user agency, and the roles that controllability and prediction errors play in this kind of trade-offs.

2.4 Automated UI Placements in AR Interfaces

Research in automated UI placements in AR mostly lies in label placements and view management [43, 50]. Little research has been conducted to explore automated placements of everyday AR UIs. In 2019, Lages and Bowman explored an adaptive walking UI, in which AR windows were placed adaptively around the user's body or on the wall based on manual input [37]. Lindlbauer et al. explored automated placements of AR content based on task and eye-tracking data [40]. Cheng et al. explored automatic adaptation of UI's spatial layouts based on environmental changes when users move to different locations [12]. Their results shed light on the potential of AR systems to predict user needs and assist the placements of AR UIs. In this research, we explore the idea of automated UI placements with different levels of automation and controllability when users move across different locations.

3 DESIGN EXPLORATION: BODYSTORM WORKSHOP

3.1 Research Goals

We first conducted a design workshop to identify user needs for accessing AR content on-the-go. The workshop was conducted online with video-conferencing software. Specifically, we aimed to: (1) identify the gaps between user needs and the existing manual UI transition mechanisms available on commercial AR systems; (2) brainstorm with experienced designers about potential solutions to address these gaps.

To design for the embodied nature of AR interactions in the space, we conducted a bodystorming workshop [54] with five UX experts. They needed to walk through their house for a sequence of physical and digital tasks. During the tasks, they experienced and reflected on a Hololens prototype we developed to represent the current UI transition mechanisms available on commercial AR devices.

3.2 Prototype for the Workshop

For the design workshop, we developed a prototype on the HoloLens 2 device. In the prototype, eight AR widgets were integrated in the system, including calendar, weather, timer, email, recipe, social, stock, and news. The widgets contain pre-defined information that we programmed in the system. All the widgets were world-referenced by default. We implemented the three common solutions for transitioning AR user interfaces. The first one was *drag&drop* (see Figure 1 (a)). Users performed a pinch gesture to grab the widgets, then they could walk to a new location and drop the widgets. The second one was *tag-along* (see Figure 1 (b-c)), in which users could touch a button to trigger the widgets to follow them around. While following, the widgets became loosely display-referenced and stayed within the field of view (FoV) of the users. By either dragging the widgets or pressing the "follow" button again, users could unfollow the widgets and make them world-fixed. Similar to how tag-along is implemented on HoloLens 2, at most one widget could be triggered to follow the users at a time. The last one was *re-instantiate* (see Figure 1 (d-e)). Users could bring out a home menu by showing their left-hand palm to the front camera of the headset. Then they could tap on the icon of a widget to re-instantiate the widget on the right side of their hands.

3.3 Bodystorm Activities

With the AR prototype application, participants were asked to go through a sequence of physical and digital tasks while moving around in their own home environments, including the kitchen, the living room, and the home office room. The tasks in the workshop were designed to represent common at-home user scenarios where digital information may be needed. Participants started in their home office. First, they were asked to place all eight widgets in their office environments. Second, participants were asked to monitor the stock widget while going to the kitchen to make a coffee. They were asked to set up a timer with the timer widget above the coffee machine. Third, participants were asked to open their fridge and check what ingredients they do not have according to the list in the recipe widget. Fourth, participants were asked to go back to the home office, record the current stock price and the missing recipe ingredients in a notepad application on their laptop. Last, participants were asked to go to the living room and check out the current weather in the weather widget. During the activities, whenever participants encountered any pain-points or challenges, they were instructed to write down notes on post-its and take screenshots with their HoloLens 2 device. In total, we allocated 25 minutes during the workshop for the bodystorming activities. Participants were also encouraged to complete more custom activities if they had time left.

3.4 Participants & Procedures

We recruited participants with rich experience in designing AR, VR, or MR user experiences in the industry, including five participants (three designers, one design technologist, and one UX researcher). All participants came from the AR/VR industry. They all had access to a HoloLens 2 hardware and were very experienced with AR/VR platforms. The workshop was conducted remotely via a
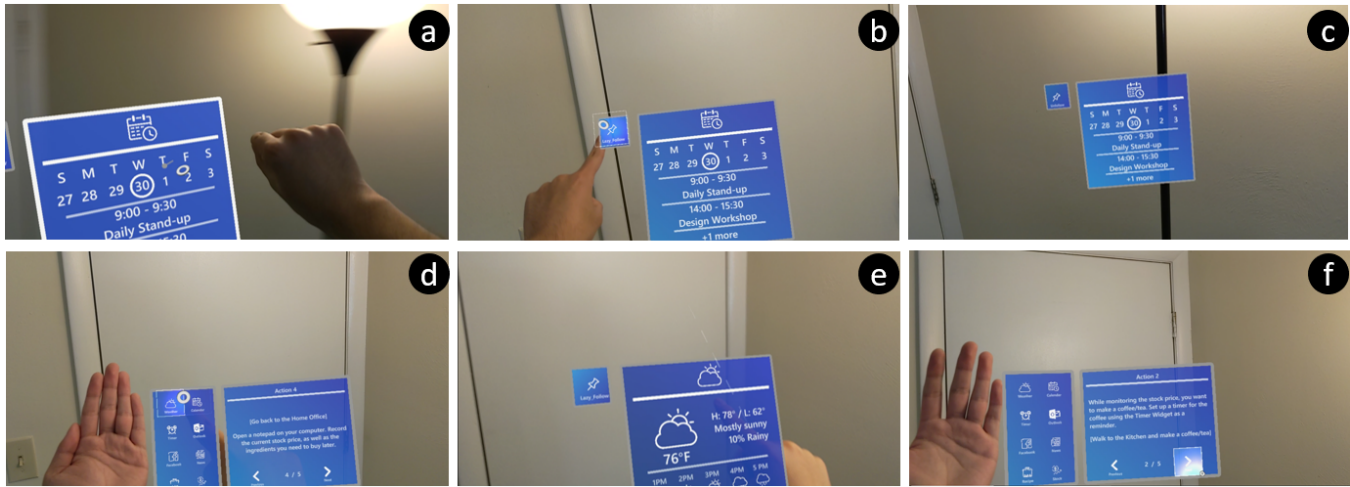
Figure 1: An illustration of the AR prototype application: (a) *drag & drop***: users approach a widget with their hands and drag them around; (b-c)** *tag-along***: users press a "follow" button beside a widget and the widget will loosely stay in the FoV of the display; (d-e)** *re-instantiate***: users press the icon of a widget in the hand menu to instantiate the widget in front of them; (f) users show their palms to see a hand menu and instruction tasks to be completed in the bodystorming session.**

video-conferencing application called BlueJeans[1]. The procedure for the workshop included seven phases. First, before the workshop, the prototype application was sent to participants together with instructions about how to sideload it on their own HoloLens 2. Second, participants joined the virtual conference room and introduced themselves to each other (5 min). Third, a brief overview was given to all participants about the workshop background and schedule (10 min). Fourth, participants were asked to go through the bodystorm activities (25 min). Fifth, after participants finished all the activities, they were instructed to go back to their home office and import the comments and screenshots in a shared online whiteboard[2] (15 min). Sixth, in the same online whiteboard board, participants were asked to brainstorm about how these pain-points could be resolved if the AR system has different levels of knowledge on their contextual changes while moving around. Each participant was encouraged to brainstorm three to five interface solutions (25 min). In the end, participants shared their solutions with each other and voted for their favourite ones. The entire workshop took around 90 minutes to complete.

3.5 Results

3.5.1 Most common pain-points. In this section, we listed the most frequently appearing pain-points mentioned by the workshop participants in the bodystorm session.

P1. Placements of the widget UIs. The first pain-points, which was mentioned by all participants, was the high level of effort required to manually place the widgets (see Figure 2 (a)). One participant commented that *"I need to spend a lot of time arranging. The widgets look messy and topsy turvy"*, and another participant commented, *"manually laying out the widgets felt tedious"*. When being asked how they wish the AR interface could be improved,

they talked about the system automation to help organize the widgets. (e.g., *"I wish the system can help me organize the widgets so they don't take up too much space around me"*).

P2. Awareness & Recall. The second pain-point, mentioned by three participants, was the difficulty of memorizing where and why a widget was placed beforehand in the previous environment (see Figure 2 (b)). One participant commented that *"I felt confused about why the timer was opened in the office"*; and the other one mentioned that *"I forgot where I placed a widget, so I had to scan the whole area around my screen"*. When being asked about improvements, one designer mentioned that *"I wish there were some guidance about where I placed what widget within my field of view"*, and the other one mentioned that *"(I want to be more aware) of the locations of the widgets, what's opened, and how I'm bringing a widget from place to place"*.

P3. High effort of widget acquisition. The last pain-point, which was mentioned by three participants, was the high-level user effort required to access the widgets. Participants mentioned that they did not want to find the widget they placed in the previous location. They wish they could easily bring multiple widgets with them and have access to certain information without the need to relocate or reopen a widget. For example, one commented that *"I wish I didn't have to reload the weather app just to check the weather"* (see Figure 2 (c)); and the other one commented that *"Only one widget following is too little. I would like to put 'quick' widgets such as weather and timer on to my forearms"*. Lastly, participants talked about leveraging the contextual change for the UI transition, for example, one asked *"Could the weather widget appear as I am walking to the door? Or as I am about to head out?"*

3.5.2 Most common interface solutions. In this section, we highlight the most frequently appearing solutions mentioned by the workshop participants in the brainstorming session.

Figure 2: Three examples from the screenshots taken by the workshop participants, each highlights one of the pain-points: (a) *P1.* placements of the widget UIs, in which participants wish that the AR system could help them arrange the UIs around the physical monitor; (b) *P2.* awareness & recall, in which one participant forgot why the timer widget was placed in the kitchen after returning there from the office; (c) *P3.* high effort of widget acquisition, in which a participant wish that did not have to reopen the weather widget and manually place it in the living room just to access the weather information.

S1. Wrist-based glanceable UIs. The first interface solution, voted by four of the participants (80%), aimed to solve **P.3** and required a low level of contextual understanding with some level of input required from the users. In this solution, designers suggested that all widgets were shown in low level-of-detail (LoD) icons and attached to the user's wrist and stays with the users by default similar to smartwatches (*"I would like to put 'quick' widgets such as weather and timer on to my forearms."*) Users could easily glance at the icons, or open the widget in full size if they need high LoD information.

S2. Snap to planes or objects. The second interface solution, voted by three of the participants (60%), aimed to solve **P.1** and **P.2**. It required some level of knowledge of the environments. In this solution, the widgets automatically snap to physical surfaces and planes, or near relevant objects after being opened. Participants suggested that *"widgets should align/snap to my physical surrounding or other already placed widgets";* and *"widgets understand what is around it and adapt to the environment".*

S3. Everything in the right place. The third interface solution, voted by four of the participants (80%), aimed to solve all the three pain-points. It required a high level of contextual understanding of the environment. In this solution, the system automatically opened the widget and placed it right when and where the users needed it. Participants suggested that *"the UI populates in the right place where it is most relevant, where the user's attention is, what the user's intention might be"*; and *"the UI should be displayed around the system's best guess of the object or activity it's related to, the user can move this."*

In general, through the workshop, we learned about the user needs and challenges when trying to use AR interfaces while carrying out real-world tasks. New interface solutions need to be explored for solving the pain-points mentioned above. The design workshop highlighted some of the potential directions, such as reducing the effort to remember, carrying AR content, and leveraging the system's contextual awareness to trigger certain widgets. We designed multiple UI transition mechanisms in sketch based on the workshop learnings.

4   EVALUATION: USER STUDY

Inspired by the solutions we generated from the design workshop, we implemented three interfaces (*Wristpack*, *Semi-Auto*, *Fully-Auto*),

which incorporated different levels of automation and controllability. We chose these three because they represent the three automation levels proposed by Findlater and McGrenere: *adaptable*, *mixed-initiative*, and *adaptive*, with an increased level of system automation and decreased level of user control. [23]. We also simulated error/inaccuracy that is unavoidable in any of the prediction-based automation algorithms.

To evaluate and understand how these interfaces are used while people move in spaces, we conducted a user study. Due to challenges of running in-person studies during COVID-19 and to avoid technological limitations of current AR devices, the study was conducted in a VR-simulated AR environment.

4.1   Research Questions

In this study, our goal was to evaluate and compare four conditions (*Wristpack*, *Semi-Auto*, *Fully-Auto*, *Baseline*) for transitioning AR widget UIs. Specifically we aim to answer the following questions through the user study:

- How do these interface transition mechanisms perform in terms of efficiency, usability, workload, and agency?
- How do different levels of automation and controllability affect agency and users' preference?
- How do people perceive and handle system inaccuracy/error when it occurs? How is the overall experience affected by the errors?

4.2   Interface Conditions

4.2.1   Wristpack. The first interface solution took inspirations from **S.1**. When users leave the current room and head towards the other one, all the widgets become automatically attached to user's wrist and forearms, displaying as icons and names. In previous work, Harrison et al. explored projecting UIs on the wrist for interacting with menus [30]. Grubert et al. explored extending wrist-worn displays with widget UIs for convenient access to mobile applications [28]. Similarly, in our *Wristpack* solution, when users need access to a widget later, they can pull the widget off their arms and place them around in the new location (see Figure 3 (a-b)). This interface represents the *adaptable* metaphor from Findlater and McGrenere's work [23], in which users take most of the control about when and where the widgets are placed in the real world; the system only
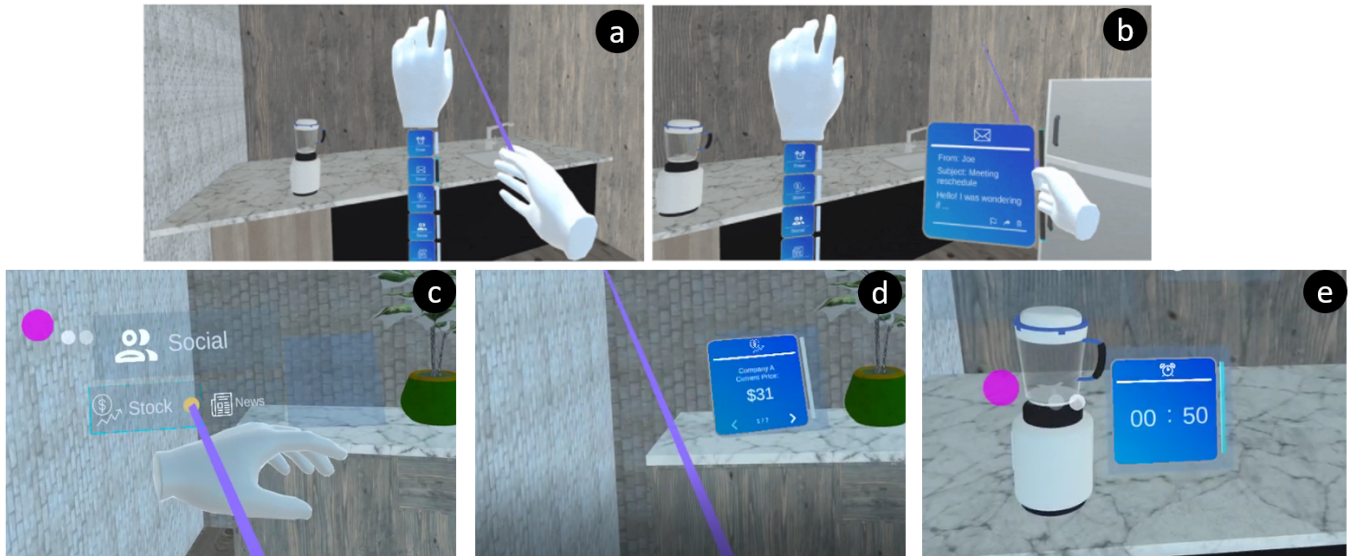
Figure 3: An illustration of the three interfaces: (a-b) *Wristpack*, in which all widgets UIs are attached to participants' wrist when spatial changes are detected and can be "pulled out" if needed; (c-d) *Semi-Auto*, in which the system would suggest three widgets on participants' wrist, and participants hold the decision of when and which one to place; (e) *Fully-Auto*, in which the system would place the most recommended widget automatically for the users without any input needed (a purple dot was displayed at the corner of the display as a visual indicator when automation happened in the system).

provided a small amount of automation when a spatial change was detected.

4.2.2 Semi-Automated Placements (Semi-Auto). The second interface solution took inspiration from **S.1**, **S.2** and **S.3**. The system would predict user needs and suggest three widgets on the user's wrist to be placed in these spaces, so it has a higher level of automation as compared to the *Wristpack* interface. The three widgets will be spawned on users' wrist, the widget with the largest probability of being needed had the biggest size and was the most visible, while the widgets that were less possibly needed was smaller and less visible (see Figure 3 (c)). The user has high controllability by making the final decision of which widget to open and when to place it (see Figure 3 (d)). If a prediction error happened, meaning that the top-recommended widget was not a match to the task, the user can look through the rest of the two less prominent recommendations and find the correct widget. This interface represents the *mixed-initiative* metaphor [23], in which the user and the system take shared control over which and when the widgets are placed in the real world.

4.2.3 Fully-Automated Placements (Fully-Auto). The third interface solution was similar to the *Semi-Auto* condition, but with a higher automation level and lower controllability level, in the sense that the user could not interfere with the system automation results. As such, a higher cost was introduced while the prediction was wrong because users were not allowed to make any change to system predictions. After predicting the widget that the user may need for a new task, the system would automatically place the top-recommended widget in front of the user without any input from the users (see Figure 3 (e)). When an error happened in the

predictions, the user needed to find the previous location of the widget to access it. The *Fully-Auto* condition represents the *adaptive* metaphor [23], in which the system takes full control over which, when and where the widgets are placed. The users could not change the predictions made by the system even when it is incorrect.

4.2.4 Baseline. We also included a *Baseline* interface, which was the drag & drop and tag-along behaviors of the widgets. Similar to the AR prototype, participants could either grab a widget with their controller and drop it at a new room, or trigger tag-along mode so the widget would follow them around automatically. Note that we removed the re-instantiation function in the VR study because: (1) we wanted to keep computational resource allocation consistent for all interface conditions; (2) our research focus was about interface transition rather than interface initiation.

4.3 Study Design

We conducted a within-subject user study to experience the above four conditions of UI transition mechanisms in VR. Due to the limitation of not being able to run in-person studies during the COVID-19 pandemic, we simulated the AR interfaces in a VR environment so that we could recruit from a larger pool of VR headset owners who have access to consumer VR hardware. The study was conducted remotely and unsupervised.

4.3.1 System. The experiment used a simulated AR setting implemented in a VR system, to avoid the limitations of current AR devices (e.g., limited FoV, unstable wide-area multiple-room tracking), to allow us to systematically control key features of the environment and task, and to potentially recruit from a larger base of participant pool online. This approach, known as Mixed Reality
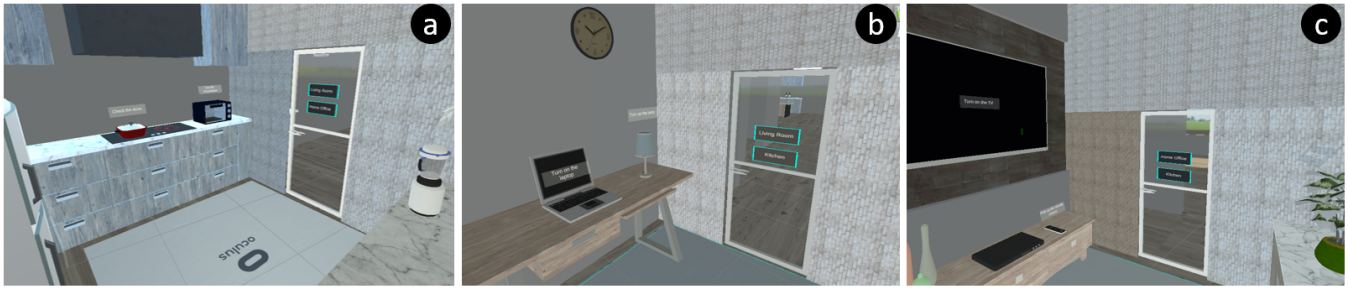
Figure 4: An illustration of the virtual home environment with three rooms: (a) the kitchen; (b) the home office; and (c) the living room. Participants could touch the buttons with the blue outlines on the door to travel between the three rooms. Each room has four usable objects, yielding a total of 12 objects in the virtual home (kitchen: stove, microwave, blender, fridge; office: laptop, lamp, bookshelf, smartphone; living room: TV, plant, remote control, trash bin).

Simulation, has been used in a variety of prior AR experiments and was proven to be effective [6, 25, 38, 39]. The Oculus Quest 2 device was used for the implementation of the VR experience. The device has 1832×1920 resolution per eye with 90 Hz refresh rate. The Oculus Touch controllers were used for interactions with the widgets in the VR environment. The experimental software was developed via Unity 2020.3.16f1 with the SDK provided by Oculus.

4.3.2 Virtual Environment. In the task, participants were placed in a virtual home environment with three rooms, the kitchen, the living room, and the home office (see Figure 4). Each room has a 2 by 2 meters walking area for participants to freely move around. In case that participants do not have access to a large enough walking area, we implemented a teleportation technique so participants could teleport in the same room. To move between the three rooms, participants could move to the virtual door and touch the corresponding button with the controllers (see Figure 4). The virtual scene would then be switched to the new room. As such, we were able to reuse the same walking space in the real world for interactions in different virtual rooms. In each room, there were four "usable" objects (see Figure 4). Users could use a raycasting technique to point the right controller at the objects to use them.

4.3.3 Tasks. A within-subject design was used for the study, in which *interface condition* was the only independent variable. Latin-square counterbalancing was applied to the order of conditions. In the task, participants were instructed to imagine that they were the owner of the virtual home. They wanted to interact with various objects in the three different rooms, for example, use the stove in the kitchen, use the laptop in the home office, or turn on the TV in the living room. A total of twelve objects were scattered in the home environment, four in each of the three rooms (see Figure 4). They were asked to move between the three virtual rooms in order to use these objects. After they interact with an object, they suddenly wanted to check some information. For example, they wanted to know the ingredients needed in the recipe after opening the fridge, or know the next calendar event after turning on the laptop. Similar to the prototype application in the design workshop, eight widgets were integrated in the system. A multiple-choice question popped up near the object simulating their thoughts of mind, and they needed to check the information in the widgets to

answer the questions. As such, in a single trial, participants were asked to go to a different room (see Figure 5 (a)), interact with a virtual object by pointing the ray at the object and press the trigger button (see Figure 5 (b)), and answer the questions prompted on the object about information in a widget (see Figure 5 (c)). For the *Semi-Auto* and the *Fully-Auto* conditions, the automation results were dependent on the questions asked (i.e., which widget was needed by the users in order to answer a question). A total of 12 trials were included for each interface. Participants were asked to answer the questions as fast as possible while prioritizing accuracy. As such, our setup simulates a scenario in which users are in a hurry and want to obtain the information they need quickly and efficiently.

4.3.4 Simulation of predictability & accuracy. Predictability and accuracy are two important aspects of adaptive UIs. According to Gajos et al., accuracy refers to "the percentage of time that the necessary UI elements are contained in the adaptive area", and predictability refers to "if the adaptation follows a strategy the users could easily model in their heads [27]." Our tasks setup simulated high predictability because users know that the system's recommended widget(s) will appear after they interact with an object during *Semi-Auto* and *Fully-Auto* conditions. Since it is extremely challenging for adaptive interfaces to reach 100% accuracy on predicting user intent, we simulated imperfect accuracy in both the *Semi-Auto* and *Fully-Auto* interface conditions. For the *Semi-Auto* condition, the system would suggest the widget needed to answer the question in the second or third slots 25% of the time (3 out of 12 trials), which posed a low cost on the users to retrieve the correct widget when imperfect prediction happened. For the *Fully-Auto* condition, the system would place the incorrect widget in front of the users 25% of the time. If that happens, since users could not interfere with system automation results, they needed to manually find the widget and access the information in it similar to the *Baseline* condition, which posed a high cost on the users to correct the prediction errors. While we understand that 25% error rate is relatively high for regular well-trained machine learning classifiers, the value is close to the accuracy levels of state-of-the-art predictive systems that predict user's interaction intent in order to provide the relevant apps, tools, and information at the right time [11, 32, 46, 59, 66], which is the use case we are targeting by
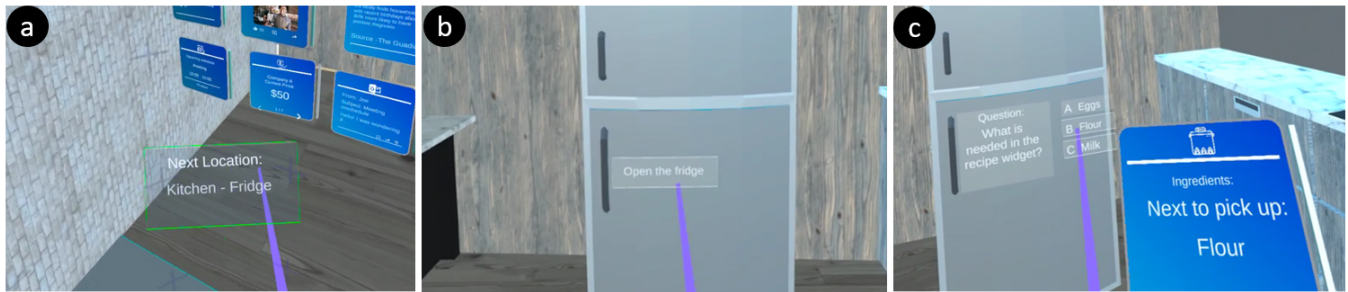
Figure 5: An illustration of a single task trial: (a) participants followed an instruction board and went to a different room to use an object; (b) participants used the object by pointing the ray at the object and press the trigger button; and (c) a multiple-choice question popped up on the object about a widget, and participant accessed information in that widget and answered the question by selecting an option.

transitioning the right widget to the right place right when the users need them. For example, Huang et al. proposed a system that predicts which app the user will open on mobile phones based on contextual information. The system prompted three apps, the hit rate of which by the users fell between 67% to 79% maximum [32]. Qu et al. compared different machine learning algorithms for predicting user intent in information seeking with conversation assistants, the accuracy levels of which lie between 63% to 69% [46]. Xia et al. proposed IntentCapsuleNet-ZSL, a zero-shot deep neural network classifier for predicting everyday interaction intent such as play music or get weather information, the accuracy levels of which fell at 75.87% the lowest [66]. Chen et al. proposed a reinforcement learning system for predicting user's query intent in automated customer service, in which the hit rate of the top three instance in the recommended list reached 75.95% [11]. As such, 25% could be an ideal simulated error rate value in order to make our results relevant to the current technological contexts in predicting user's intent specifically in everyday interactions such as information seeking or opening an app.

4.3.5 Procedure. The study was completely remote and unsuper-vised on the dscout platform[3]. The study includes six phases. In the first phase, a screener questionnaire was sent out to the dscout platform. Participants were required to have access to the Oculus Quest 2 hardware, internet connection, and at least a 5 by 5 feet area to move around safely. Second, qualified participants were invited to the project on dscout, which granted them access to the test software and the questionnaire. Participants were instructed to complete a background questionnaire, and install the test soft-ware on their own Quest device. Third, participants opened the test application. The application started with a tutorial about the environment, controls, and tasks. Fourth, participants experienced the four interface conditions one by one. Before the formal testing session of each interface condition, a training session was provided to participants in VR to teach them how to use the interface. After they finish the 12 trials for each interface, they were instructed to take off the VR headset, go to their laptop and complete one page of the questionnaire on dscout. The questionnaire asked about the usability, workload, agency, as well as what they like and dislike

about the interface in the condition they just experienced. Fifth, after participants finished all the four interfaces, they clicked on an upload button to upload the logged data to a cloud server. Last, participants were instructed to rank the four interfaces based on their own experience. They were also asked about how they felt when the system suggested the wrong widgets in the tasks. The study took about 80 minutes in total. Participants are compensated with $70 US dollars for their time. To encourage participants to achieve as good performance as they can, we rewarded half of the participants who performed more accurately and faster than the other half another $10 dollars.

4.3.6 Participants. Participants were recruited from the dscout platform. The dataset includes 40 participants (23 M, 17 F) between 18 to 55 years old (M = 34.53, SD = 9.33). All participants had prior experience with VR and were regular users of the Oculus Quest 2 device.

4.4 Measures

4.4.1 Performance measures. For evaluating user performance on the tasks, we calculated (1) the time of completion (how long did it take for participants to finish a task); (2) the distance travelled (including distance teleported and distance walked); as well as (3) the accuracy of the answers for each interface condition.

4.4.2 Subjective Measure. We used the Single Easement Question-naire (SEQ) [53], System Usability Scale (SUS) [5] and NASA TLX workload questionnaire [31] to gauge the usability, effectiveness and workload of each interface condition. We also asked partic-ipants to rate the level of agency on each interface using three questions adapted from the work by Tapal et al. (see Figure 9 (b)) [58], and rank the interface based on their own preferences.

4.5 Results

We conducted a series of analyses to our results. Shapiro-Wilk test indicated that both the time and distance were not normally distributed for *Wristpack*, *Semi-Auto* and *Fully-Auto*. As such, we applied Box-Cox transformations to correct non-normal residuals [4, 7, 60, 69], followed by Repeated-Measure One-way ANOVA (RM-ANOVA) tests to reveal the main effect of independent vari-ables. A Greenhouse-Geisser correction was applied for violations

---

[3]https://dscout.com/

of sphericity. For Likert measures, Friedman tests were applied with Wilcoxon signed-rank test as post-hoc pairwise analysis. The Pearson's correlation coefficient r was reported as a measure of effect size [9, 49]. According to Cohen's measure [16, 17], $0.1 \leq r < 0.3$, $0.3 \leq r < 0.5$, and $r \geq 0.5$ would be considered as small, medium and large effects respectively. Bonferroni correction was applied to all pairwise comparisons. We used an α level of 0.05 in all significance tests. In the result figures, pairs that are significantly different are marked with * when $p \leq .05$, ** when $p \leq .01$ and *** when $p \leq .001$.

4.5.1 Performance measures.
In this subsection, results about the performance measures are reported in detail.

Time. A Box-Cox transformation with $\lambda = 0.3$ was applied to correct non-normal residuals. RM-ANOVA indicated significant main effect on *interface* on the average time it took for participant to answer each question ($F(2.017, 78.680) = 44.556, p < .001, \eta_p^2 = .533$). Post-hoc pairwise comparisons with Bonferroni adjustments indicated that *Baseline* yielded significantly more time to answer the question as compared to the *Wristpack* (p < .001), *Semi-Auto* (p < .001), and *Fully-Auto* (p < .001) conditions. *Semi-Auto* also took significantly less time as compared to *Wristpack* (p < .001) and *Fully-Auto* (p < .001) (see Figure 6 (a)).

Figure 6 (b) shows the average time it took for participants to answer the questions under four *scenarios*: when the system prediction was correct or wrong for *Semi-Auto* or *Fully-Auto* interfaces. A Box-Cox transformation with $\lambda = -0.6$ was applied. RM-ANOVA yielded significant main effect of *scenario* on the average time ($F(3, 117) = 440.201, p < .001, \eta_p^2 = .919$). Post-hoc pairwise comparisons indicated that when a prediction error happened during the *Fully-Auto* condition, participants spent significantly longer time to answer the questions as compared to when error happened in the *Semi-Auto* condition ($p < .001$), as well as when the predictions were correct in *Semi-Auto* ($p < .001$) and *Fully-Auto* ($p < .001$). No difference was found for *Semi-Auto* between when the system suggested the correct and when the system suggested the wrong AR content ($p = .086$). This result shows that in the *Fully-Auto* condition, users spent more time handling the prediction error, as compared to the *Semi-Auto* condition.

Distance-travelled. A Box-Cox transformation was applied with $\lambda = 0.3$. RM-ANOVA also found significant main effect of *interface* on the total distance travelled ($F(2.256, 87.980) = 77.528, p < .001, \eta_p^2 = .565$) (see Figure 7 (a)). Post-hoc pairwise analysis with Bonferroni adjustments showed that *Baseline* yielded significantly more distance travelled as compared to the *Wristpack* (p < .001), *Semi-Auto* (p < .001), and *Fully-Auto* (p = .002) conditions. The *Fully-Auto* condition also resulted in significantly more distance travelled as compared to the *Wristpack* (p < .001) and the *Semi-Auto* conditions (p < .001).

Accuracy. Figure 7 (b) shows the average accuracy level of the answers for each of the interface conditions. We processed the accuracy data with Aligned Rank Transform (ART) to take into account the non-normal distributions [4, 63, 69]. RM-ANOVA yielded significant main effect of *interface* on accuracy rate ($F(3, 117) = 7.712, p < .001, \eta_p^2 = .165$). Post-hoc comparisons show that the

Baseline condition resulted in significantly lower accuracy as compared to *Wristpack* ($p < .001$), *Semi-Auto* ($p = .001$), and *Fully-Auto* ($p = .002$) conditions.

4.5.2 Subjective measures.
In this subsection, results about the subjective measures are shown in detail.

User preference. Figure 8 (a) shows the distributions of the subjective rankings. 27 our of 40 (72.5%) participants ranked *Baseline* as the least favored interface. 31 participants (77.5%) ranked *Semi-Auto* as the most favored interface. 21 participants (52.5%) ranked the *Fully-Auto* condition as the second most favored interface.

SEQ. Figure 8 (b) shows participants' response to the SEQ. Friedman test yielded significant main effect of *interface* on the ratings ($\chi^2(3) = 22.610, p < .001$). Wilcoxon signed-rank tests showed that the *Semi-Auto* condition was rated significantly higher than the *Baseline* condition (Z = -3.408, p = .003, r = .381). No significant differences were identified between other pairs.

Usability. Figure 9 (a) shows participants' responses towards three questions in the SUS questionnaire. Friedman tests found significant main effect of *interface* on all the three statements (**U1: I thought the interface was easy to use**; **U2: I would imagine that most people would learn to use the interface very quickly**; and **U3: I felt very confident using the interface**) (all $p < .001$). For **U1**, *Semi-Auto* was rated significantly higher than *Baseline* (Z = -4.979, p < .001, r = .556) and *Wristpack* (Z = -4.861, p < .001, r = .543). *Fully-Auto* was also rated significantly higher than *Baseline* (Z = -3.953, p < .001, r = .442) and *Wristpack* (Z = -3.484, p = .002, r = .390). Similarly, for **U2**, *Semi-Auto* was rated significantly higher than *Baseline* (Z = -4.401, p < .001, r = .492) and *Wristpack* (Z = -3.714, p < .001, r = .415). *Fully-Auto* was also rated significantly higher than *Baseline* (Z = -3.399, p = .004, r = .496) and *Wristpack* (Z = -2.855, p = .042, r = .319). For **U3**, *Semi-Auto* was rated significantly higher than *Baseline* (Z = -4.679, p < .001, r = .523), *Wristpack* (Z = -4.438, p = .001, r = .496), and *Fully-Auto* (Z = -3.714, p = .024, r = .415).

Agency. Figure 9 (b) shows participants' responses to the three questions regarding agency (**A.1: To what extent did you feel the decision of where and when to place a widget was within your hands; A.2: To what extent did you feel the widgets were placed with your intent; A.3: I felt that I am responsible for the speed and accuracy of completing the task**). Friedman test yielded significant main effects of *interface* on the ratings for all the three questions (all $p < .002$). Participants found that the decision of where and when to place the widgets was significantly less in their hand for the *Fully-Auto* condition as compared to *Baseline* (Z = -4.800, p < .001, r = .537), *Wristpack* (Z = -4.214, p < .001, r = .471), and *Semi-Auto* (Z = -3.846, p < .001, r = .429). When being asked to what extent did they feel that the widgets were placed with their intent, *Fully-Auto* was rated significantly lower as compared to *Baseline* (Z = -3.153, p = .012, r = .352) and *Semi-Auto* (Z = -3.707, p < .001, r = .414). When being asked how much they feel that they are responsible for the speed and accuracy for completing the tasks, *Semi-Auto* received significantly higher ratings than *Baseline* (Z = -3.063, p = .012, r = .342), *Wristpack* (Z = -3.375, p = .006, r = .377). and *Fully-Auto* (Z = -4.893, p < .001, r = .547).
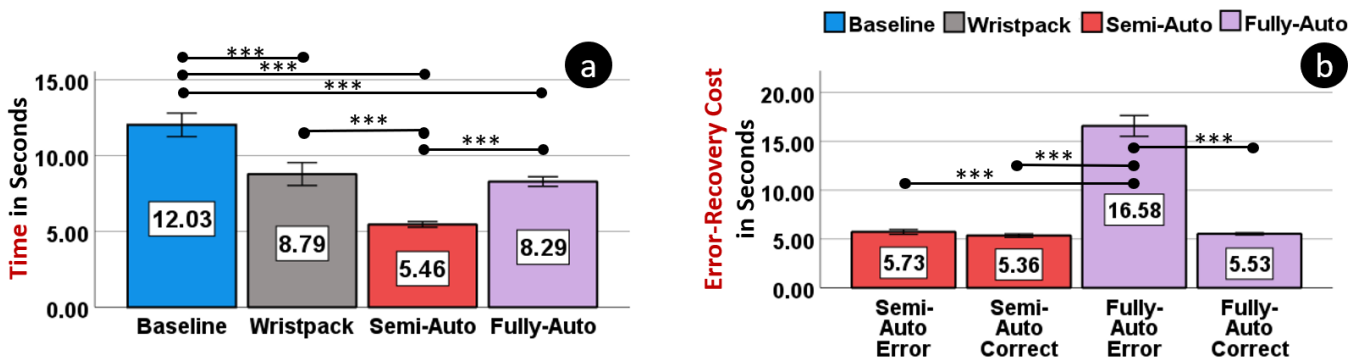
Figure 6: (a) The average time it took for participants to answer each question (in seconds); (b) the average time took for participants to answer the questions when failure happened / did not happen in the prediction for *Semi-Auto* and *Fully-Auto* interfaces (in seconds) (±*S.E.*).
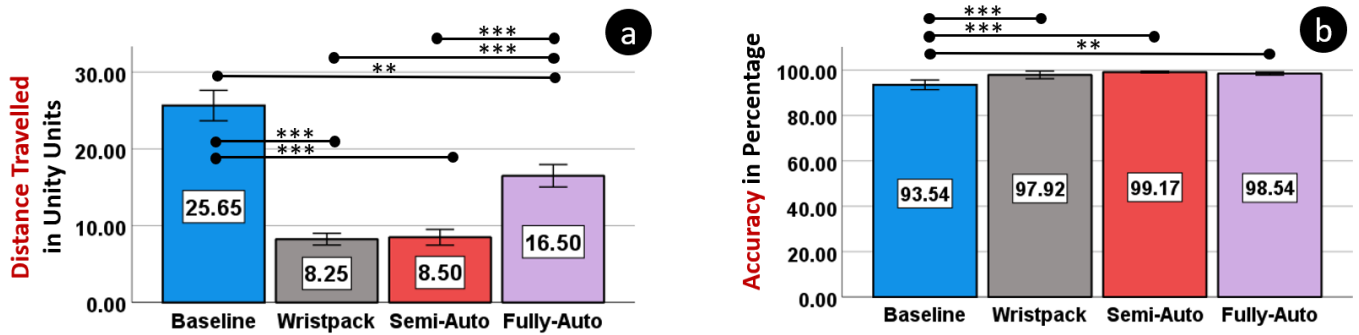

Figure 7: (a) The average distance travelled by participants to answer each question (in Unity units); (b) the average accuracy level of the answers for each interface condition (in percentage) (±*S.E.*).
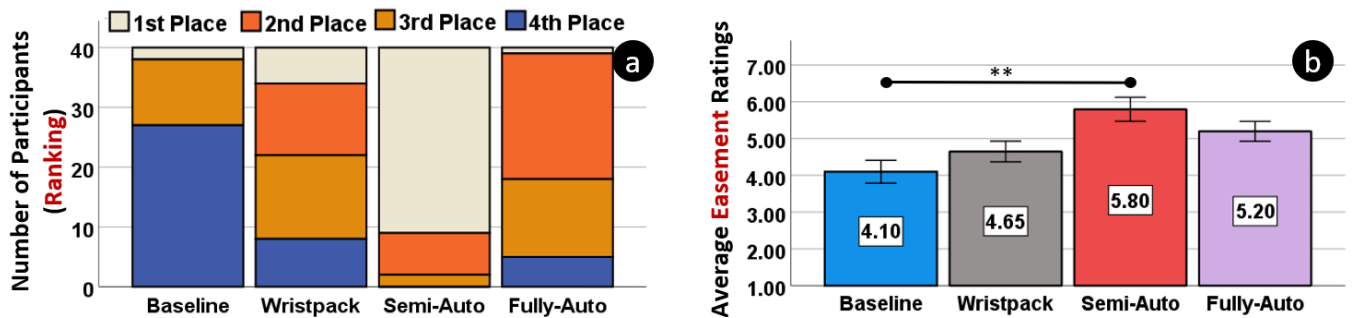

Figure 8: (a) The ranking distributions for each interface; (b) the average ratings of the SEQ questionnaire (±*S.E.*).

Workload. Figure 10 shows the NASA TLX ratings for five categories. Pairwise comparisons showed that the *Baseline* condition yielded a significantly higher level of mental workload and effort than *Wristpack*. It also resulted in a significantly higher level of mental, physical, effort, and frustration as compared to *Semi-Auto* and *Fully-Auto*. The *Semi-Auto* condition resulted in a significantly lower level of mental, physical, effort and frustration as compared to *Wristpack*. Meanwhile, it also yielded a lower level of effort and frustration than the *Fully-Auto* condition.

4.5.3 Qualitative feedback. To understand why participants liked or disliked the interfaces, we collected qualitative feedback by asking participants to comment on what they like and dislike about each interface. Below we highlight most commonly appeared comments by participants.

Baseline. Participants liked the *"intuitiveness"* and *"the sense of being in control;"* they disliked that it was *"sometimes cumbersome and slow,"* as shown in the quotes like, *"I always have to travel to*
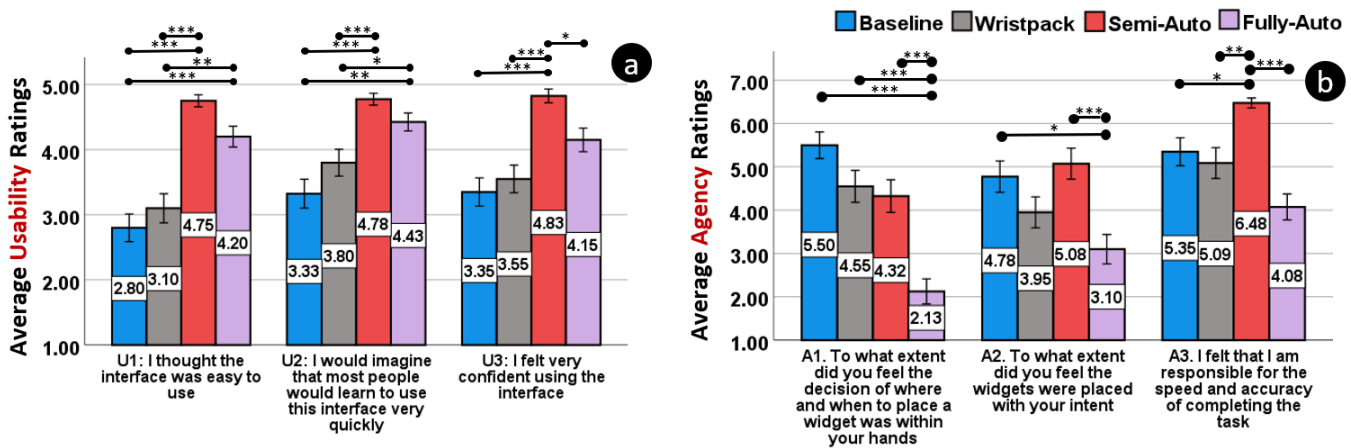
Figure 9: (a) The average ratings for the three questions from the SUS questionnaire; (b) the average ratings to three questions about agency for each interface condition (±*S.E.*).
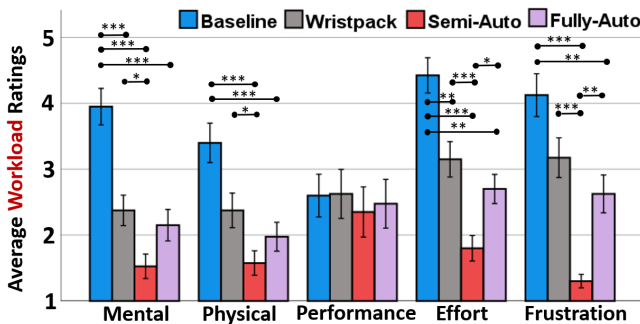

Figure 10: The average workload ratings from NASA-TLX subcategories (±*S.E.*).

get the widget I need," and "I need to recall where I placed the widgets beforehand to find them."

Wristpack. Participants liked the *"convenience," "similarity to smart watch,"* and *"no need to think about carrying the widgets with them;"* they disliked that they *"had to search for the correct widget on their wrist to open."*

Semi-Auto. Participants liked the *"ease of use, accuracy."* Moreover, they liked *"the sense of being in control,"* as witnessed in quotes like *"I really liked that I was completely in control of the widgets that I wanted to see up here,"* and *"it still gives you the convenience of having things pop up, but then you can control and make sure you're getting the correct one."* They also liked they were able *"to find the right widget in the recommended list, even when it was not the top one."* They disliked that they *"can't easily select another widget if they accidentally picked the wrong widget."*

Fully-Auto. Participants liked the *"the prediction was correct most of the time"* and *"it was absolutely awesome when it worked;"* they disliked *"not being able to correct the widget when failure happens"* and *"high effort to correct the error when it happens by manually finding the widget they need."*

4.6 Summary of Findings

In summary, we evaluated four mechanisms for UI transitions for tasks that require users to move in spaces. These mechanisms include: *Wristpack* (users carry the widgets on the wrist when moving from one space to another), *Semi-Auto* (the system predicts three widgets that may be needed for the task, and the user makes the final choice, in which the cost of automation error is low), *Fully-Auto* (the system predicts what widgets may be needed for the task and presents the top one directly to the user, in which the cost of automation error is high) and the *Baseline* conditions (the widgets need to be manually moved or tethered by the user). We found:

- The *Semi-Auto* condition performed the best both objectively (time of completion and traveled distance) and subjectively (user preference, workload, usability, and agency) among all four conditions;
- The *Baseline* condition performed the worst among all conditions;
- The participants felt significantly less agency during *Fully-Auto* condition than the more manual conditions (i.e., *Baseline* and *Wristpack*). In contrast, participants felt an equal or even higher level of agency on *Semi-Auto* condition as compared to the manual conditions.
- When a prediction error happened, users spent a lot shorter time in handling the error in the *Semi-Auto* condition than in the *Fully-Auto* condition.
- From the qualitative feedback, we found that users considered *the sense of control* and *ease of recovering from error (could be from user error or system's prediction error)* as the key factors when deciding their preference.

4.7 Discussion & Design Implications

4.7.1 The need for ultra-low-friction interfaces on-the-go. Our results provided strong evidence that the current mechanism (i.e., manual movement of the AR widgets) was not optimal for transitioning widget UIs spatially. It was the least preferred interface for most participants, resulted in a lower level of accuracy and

efficiency, and posed high workload on the users. The major reason was that in *Baseline*, users had to remember which information was needed, recall where the widgets were located in the previous environment, and manually acquire them in order to answer each question. The heavy mental and physical workload made it challenging for the users to obtain the correct answers. The *Wristpack* interface offloaded part of the workload by carrying the widgets automatically on user's wrist. However, it was still not optimal in that the users need to manually locate the widget on their wrist, open it, and place it in the new location. For scenarios that the users move around in different spaces to carry out different tasks, users are already multitasking - they navigate the space, look for different physical objects, and sometimes handle social encounters. When users need digital content in such scenarios, they have less cognitive bandwidth to maneuver UI widgets, therefore needing the ultra-low-friction interface mechanisms. In both our design workshop and user study, we confirmed that this user need does exist, calling for more solution explorations from the HCI research and design community.

4.7.2 Automation, controllability and agency. One of the motivations of our work is to explore how automation and controllability levels can make a difference in addressing the dynamic UI needs on-the-go. We designed the *Wristpack*, *Semi-Auto* and *Fully-Auto* interfaces to integrate different levels of automation and user control. In *Wristpack* condition, the previously opened widgets automatically minimize and attach to the wrist when major spatial differences are detected. In *Semi-Auto* condition the system automatically finds a list of matching widgets for the task. In *Fully-Auto* condition the system automatically places the best matching widgets for the task. All three interfaces were able to reduce the workload and increase the accessibility of the widgets. However, *Semi-Auto* condition, where the system suggests a few widgets for the task and the user makes the choice, did the best objectively and subjectively. Clearly, under situations where errors are inevitable and happen 25% of the time, more automation does not necessarily lead to better user experiences. Along the same line with the previous work, we confirm the importance of controllability, which is how much the user is in control of an automated task [51, 56]. Moreover, our results confirmed that a combination of automation and controllability creates the best user experience outcomes for tasks on-the-go. Controllability also plays a critical role in user agency. In our study, the *Fully-Auto* condition had lower agency ratings than the more manual conditions (i.e., *Baseline* and *Wristpack*). But we also found that participants felt an equal or even higher level of agency on *Semi-Auto* condition as compared to the more manual conditions. This result, combined with users' positive comments around controllability, indicates that giving users the control for decision-making can keep the agency high while leveraging system automations.

We would also like to highlight the relevance of our findings with previous work in predicting typing intent during text entry. While entering texts, keyboards with predictive features could (1) recommend a list of words (usually 2-3 words on mobile interfaces) based on what is already typed (i.e., word-prediction), the structure of which is similar to our *Semi-Auto* interface; and (2) automatically correct the typed word to another word without any input from

the users, but could happen falsely and change the typed words to undesired phrases (i.e., auto-correction), the structure of which is similar to our *Fully-Auto* condition in terms of system-level control. Previous work has found that word-prediction could reduce the required keystrokes by giving users both automation and control over world selections [24]. However, it might also introduce extra interaction and perception costs by requiring users to pay continuous attention to the list of suggested words [47, 48]. In our work, we successfully indicated the importance of both controllability and automation by demonstrating the advantages of the *Semi-Auto* condition. However, the findings of our work were restrained to interface placements in AR in a hurried scenario with 25% errors. A higher level of controllability may introduce extra interaction and perception costs, which could outweigh the benefits brought by the automation. In different task contexts or systems, the degree level of user control and system automation needs to be carefully considered and balanced to achieve the optimal user experience.

Overall, our study confirms that automation is a promising design direction that can greatly reduce users' effort and attention cost on-the-go. Controllability is especially critical to ensure higher user agency when the system provides automation functions. Moreover, we call out for further explorations about how to combine automation with controllability. The tasks in our study were easy to combine both because the system can wait on the user to make the choices. What if the user choices are more time-sensitive (e.g. decisions when driving), how do we balance the automation and controllability in such tasks?

4.7.3 The cost of correcting prediction errors. Error has always been one of the biggest concerns for intelligent systems [45], which motivated us to study the user experience outcomes when an error occurs. In both *Semi-Auto* and *Fully-Auto* conditions, the top recommendation from the system was occasionally wrong. However, *Semi-Auto* condition is different from *Fully-Auto* condition in two ways, one is that the correct widget can be found among the other recommended items, just in a lower order and a smaller size; and the other is that users need to choose which widget to use from the recommended list. As a result of the difference, users spent significantly more time handling the prediction error during the *Fully-Auto* condition than the *Semi-Auto* condition.

While being asked about their feelings when the system suggested the wrong widget, participants gave very different responses for the *Semi-Auto* and *Fully-Auto* conditions. For the *Fully-Auto* condition, most participants mentioned "Annoying" (65%) and "Frustrated" (50%), as shown in comments like the following: *"The uncertainty of knowing if it would be right or not was very annoying and made me anxious"*; *"I was slightly annoyed because I have to find the widget I truly need, which adds lots of unnecessary work"*; and *"I would feel less frustrated if I could've grabbed the correct widget from a UI after failure happens."* In summary, a big source of frustration came from the effort of correcting the error. On the contrary, for *Semi-Auto* condition, the majority of the participants did not find it bothering when the top-recommended widget was not correct. They can easily find the correct widget from the rest of recommended list. They commented *"sometimes the widget (I need) was not at the top, which is totally fine cause I could still find it in the list;"* *"even though I had to click something other than the*

fully automated one, I could always check and get the right widget." Needing to select the widget actually gave users a good opportunity for double-checking and recognizing the recommendation error. Interestingly, if the user accidentally selected the wrong widget (i.e. user slip), they also complained about the effort they had to take to correct it, not too dissimilar to the comments about the effort required for correcting the system error during *Fully-Auto* condition.

Our findings were established on an accuracy level of 75% in a demanding scenario when accuracy was prioritized. While it is true that the users might be more tolerant of having a higher error-recovery cost when errors happen less frequently or when users are in a lightweight scenario, our results demonstrated that in worst-case scenarios where the errors happen inevitably and users are in a hurry during AR UI transitions, the cost of correcting them could play a crucial role in performance and user experience. In recent work, Lafreniere et al. proved that the temporal cost of recovering automation errors could significantly affect user frustration and experience [36]. Similarly, in previous work about the auto-correction feature in text-entry, it was found that when users have to manually correct the system's faulty auto-corrections, the cost of it could outweigh the reduced effort when the corrections were desired [1]. How much effort is needed for error recovery (including both system-generated prediction error and user error) is crucial to the user experience. The usability heuristics about "helping users recognize, diagnose, and recover from errors [44]" needed to be expanded and emphasized for today's automated/intelligence systems. Errors are not edge cases anymore, it always happens with the probabilistic output of AI systems. We need to *"always enable an easy path to recognize and recover from error."*

The error recognition and recovery could be achieved through a human-system team effort. For example, on the user side, the users could learn from the prediction errors about the limitations of the systems, thus becoming more prepared to correct the automation errors quickly. One limitation of this approach, as indicated by previous work in text-entry, would be that the users' abilities of identifying and adapting to errors vary among individuals and are largely affected by how long they have been using the system and how frequently the errors occur [2, 10]. A more reliable way would be from the system side, in which the system could incorporate functions to involve users in the loop to help it identify and learn from the prediction errors [65, 70]. Moreover, the system could even auto-detect its error based on the confidence level and users' responses/reactions.

4.7.4 The stakes of error occurrences. We would also like to point out that what is at stake when errors happen could largely affect user behaviors of using automated interfaces. Although the definition of automation level in our use case is similar to the automated driving use case, there are two major differences: (1) the consequence of an error is much less severe (lower cost of error); and (2) it is much easier to recover from the error, as the user can always find the widgets manually when the system's prediction was wrong. Our application scenario is more general-purpose and focuses on the use of AR widgets on-the-go. We do not consider our learnings here to be directly applicable to a scenario that has much higher stakes for prediction errors. Through this work, we call out to the research community about the importance of studying error from prediction algorithms with more depth and nuance.

To conclude, for the design space of AR UI transitions on-to-go, our results show that user experiences could benefit from introducing automation, such as detecting contextual changes and predicting the user intent. At the same time, we need to creatively combine automation with controllability to ensure high agency and overall satisfaction. Moreover, we should always provide an easy way for the users and the system to recognize and recover from the always-gonna-be-there prediction errors.

5 LIMITATION & FUTURE WORK

There are several limitations of our work. First of all, our study was conducted in VR due to COVID-19 restrictions and the limitations of current AR devices. Future work could evaluate the interface conditions in AR with real-world environments and tasks. Second, to ensure a safe walking environment and overcome space limitations, we implemented teleportation for locomotion in the virtual environment in the remote study. Based on recent research, teleportation may hinder spatial cognition performances as compared to real walking [13, 33]. Future research could consider involving real walking of the participants to compare these interfaces. Third, our task setup simulated a scenario that encouraged efficiency. The users were incentivized to access the widgets they needed as fast and accurately as possible. However, in everyday AR scenarios, users may access AR content at their own pace. Perhaps there will be more need for UIs that suggest non-utilitarian widgets. Future work could capture and design these AR use cases, and situate the UI mechanisms with more diverse scenarios. Fourth, we adopted a 75% accuracy level for the *Semi-Auto* and *Fully-Auto* conditions. Future research could explore how different accuracy levels could affect the user behavior and user experience of using these interfaces, and the design choices to be made. Fifth, our findings indicate that a *Semi-Auto* interface with high controllability and low error-recovery cost would likely be more favored in a hurried scenario. In future work, we plan to explore the benefits and drawbacks of having such interfaces. Last, we are interested in designing and researching lightweight/low-effort methods to recover from prediction errors, without compromising usability and agency.

6 CONCLUSIONS

In this research, we aimed to support the UI transition needs when people use AR interfaces on-the-go. We conducted an AR design workshop to reveal the existing challenges when accessing AR content across multiple spaces. We designed three interfaces to address these challenges with different levels of automation and user control. In a VR-simulated AR user study, we found that the semi-automated condition stood out as the best performing and most favored one. Thanks to the balance between automation and controllability in the semi-automated condition, user agency was not compromised when the automation level increased as compared to the manual conditions. Moreover, our study indicated the importance of error recovery cost when an error happens in predicting the exact AR content that users may need. We would like to further explore ways to fail gracefully with backup plans for

automated interfaces that rely on predictions. The findings and design insights from this work can provide valuable lessons to design ultra-low-friction AR interfaces with automation, controllability, and low error-recovery cost, especially for scenarios where users have limited attention bandwidth.

ACKNOWLEDGMENTS

REFERENCES

[1] Ohoud Alharbi, Wolfgang Stuerzlinger, and Felix Putze. 2020. The Effects of Predictive Features of Mobile Keyboards on Text Entry Speed and Errors. *Proc. ACM Hum.-Comput. Interact.* 4, ISS, Article 183 (nov 2020), 16 pages. https://doi.org/10.1145/3427311

[2] Ahmed Sabbir Arif and Wolfgang Stuerzlinger. 2014. User Adaptation to a Faulty Unistroke-Based Text Entry Technique by Switching to an Alternative Gesture Set. In *Proceedings of Graphics Interface 2014* (Montreal, Quebec, Canada) *(GI '14).* Canadian Information Processing Society, CAN, 183–192.

[3] Ronald T. Azuma. 1997. A Survey of Augmented Reality. *Presence: Teleoperators and Virtual Environments* 6, 4 (08 1997), 355–385. https://doi.org/10.1162/pres.1997.6.4.355

[4] Marc Baloup, Thomas Pietrzak, and Géry Casiez. 2019. RayCursor: A 3D Pointing Facilitation Technique Based on Raycasting. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland Uk) *(CHI '19).* Association for Computing Machinery, New York, NY, USA, 1–12. https://doi.org/10.1145/3290605.3300331

[5] Aaron Bangor, Philip T. Kortum, and James T. Miller. 2008. An Empirical Evaluation of the System Usability Scale. *International Journal of Human–Computer Interaction* 24, 6 (2008), 574–594. https://doi.org/10.1080/10447310802205776

[6] Doug A Bowman, Cheryl Stinson, Eric D Ragan, Siroberto Scerbo, Tobias Höllerer, Cha Lee, Ryan P McMahan, and Regis Kopper. 2012. Evaluating effectiveness in virtual environments with MR simulation. In *Interservice/Industry Training, Simulation, and Education Conference*, Vol. 4. 44.

[7] G. E. P. Box and D. R. Cox. 1964. An Analysis of Transformations. *Journal of the Royal Statistical Society: Series B (Methodological)* 26, 2 (1964), 211–243. https://doi.org/10.1111/j.2517-6161.1964.tb00553.x

[8] Robert Bridle, Eric McCreath, et al. 2005. Predictive menu selection on a mobile phone. In *Workshop W7 on mining spatio-temporal data, Portugal.* Citeseer.

[9] Alan Bryman and Duncan Cramer. 2009. *Quantitative data analysis with SPSS 14, 15 & 16: A guide for social scientists.* Routledge/Taylor & Francis Group.

[10] Daniel Buschek, Benjamin Bisinger, and Florian Alt. 2018. *ResearchIME: A Mobile Keyboard Application for Studying Free Typing Behaviour in the Wild.* Association for Computing Machinery, New York, NY, USA, 1–14. https://doi.org/10.1145/3173574.3173829

[11] Cen Chen, Chilin Fu, Xu Hu, Xiaolu Zhang, Jun Zhou, Xiaolong Li, and Forrest Sheng Bao. 2019. 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* (Paris, France) *(SIGIR'19).* Association for Computing Machinery, New York, NY, USA, 1265–1268. https://doi.org/10.1145/3331184.3331370

[12] Yifei Cheng, Yukang Yan, Xin Yi, Yuanchun Shi, and David Lindlbauer. 2021. SemanticAdapt: Optimization-Based Adaptation of Mixed Reality Layouts Leveraging Virtual-Physical Semantic Connections. In *The 34th Annual ACM Symposium on User Interface Software and Technology* (Virtual Event, USA) *(UIST '21).* Association for Computing Machinery, New York, NY, USA, 282–297. https://doi.org/10.1145/3472749.3474750

[13] Lucia A Cherep, Alex F Lim, Jonathan W Kelly, Devi Acharya, Alfredo Velasco, Emanuel Bustamante, Alec G Ostrander, and Stephen B Gilbert. 2020. Spatial cognitive implications of teleporting through virtual environments. *Journal of Experimental Psychology: Applied* 26, 3 (2020), 480. https://doi.org/10.1037/xap0000263

[14] Karen Church, Mauro Cherubini, and Nuria Oliver. 2014. A Large-Scale Study of Daily Information Needs Captured in Situ. *ACM Trans. Comput.-Hum. Interact.* 21, 2, Article 10 (feb 2014), 46 pages. https://doi.org/10.1145/2552193

[15] Karen Church and Barry Smyth. 2008. Understanding Mobile Information Needs. In *Proceedings of the 10th International Conference on Human Computer Interaction with Mobile Devices and Services* (Amsterdam, The Netherlands) *(MobileHCI '08).* Association for Computing Machinery, New York, NY, USA, 493–494. https://doi.org/10.1145/1409240.1409325

[16] Jacob Cohen. 1988. Statistical power analysis for the behavioral sciences. *England: Routledge* (1988). https://doi.org/10.4324/9780203771587

[17] Jacob Cohen. 1992. A power primer. *Psychological bulletin* 112, 1 (1992), 155. https://doi.org/10.1037/0033-2909.112.1.155

[18] Ashley Colley, Juho Rantakari, and Jonna Häkkilä. 2014. Augmenting the Home to Remember: Initial User Perceptions. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication* (Seattle, Washington) *(UbiComp '14 Adjunct).* Association for Computing Machinery, New York, NY, USA, 1369–1372. https://doi.org/10.1145/2638728.2641717

[19] Mary L Cummings and Sylvain Bruni. 2009. Collaborative human–automation decision making. In *Springer handbook of automation.* Springer, 437–447. https://doi.org/10.1007/978-3-540-78831-7_26

[20] Brendan David-John, Candace Peacock, Ting Zhang, T. Scott Murdison, Hrvoje Benko, and Tanya R. Jonker. 2021. Towards Gaze-Based Prediction of the Intent to Interact in Virtual Reality. In *ACM Symposium on Eye Tracking Research and Applications* (Virtual Event, Germany) *(ETRA '21 Short Papers).* Association for Computing Machinery, New York, NY, USA, Article 2, 7 pages. https://doi.org/10.1145/3448018.3458008

[21] David Dearman, Melanie Kellar, and Khai N. Truong. 2008. An Examination of Daily Information Needs and Sharing Opportunities. In *Proceedings of the 2008 ACM Conference on Computer Supported Cooperative Work* (San Diego, CA, USA) *(CSCW '08).* Association for Computing Machinery, New York, NY, USA, 679–688. https://doi.org/10.1145/1460563.1460668

[22] S. Di Verdi, D. Nurmi, and T. Hollerer. 2003. ARWin - a desktop augmented reality Window Manager. In *The Second IEEE and ACM International Symposium on Mixed and Augmented Reality, 2003. Proceedings.* 298–299. https://doi.org/10.1109/ISMAR.2003.1240729

[23] Leah Findlater and Joanna McGrenere. 2004. A Comparison of Static, Adaptive, and Adaptable Menus. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Vienna, Austria) *(CHI '04).* Association for Computing Machinery, New York, NY, USA, 89–96. https://doi.org/10.1145/985692.985704

[24] Andrew Fowler, Kurt Partridge, Ciprian Chelba, Xiaojun Bi, Tom Ouyang, and Shumin Zhai. 2015. Effects of Language Modeling and Its Personalization on Touchscreen Typing Performance. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems* (Seoul, Republic of Korea) *(CHI '15).* Association for Computing Machinery, New York, NY, USA, 649–658. https://doi.org/10.1145/2702123.2702503

[25] Joseph L. Gabbard, J. Edward Swan, and Deborah Hix. 2006. The Effects of Text Drawing Styles, Background Textures, and Natural Lighting on Text Legibility in Outdoor Augmented Reality. *Presence* 15, 1 (2006), 16–32. https://doi.org/10.1162/pres.2006.15.1.16

[26] Krzysztof Z. Gajos, Mary Czerwinski, Desney S. Tan, and Daniel S. Weld. 2006. Exploring the Design Space for Adaptive Graphical User Interfaces. In *Proceedings of the Working Conference on Advanced Visual Interfaces* (Venezia, Italy) *(AVI '06).* Association for Computing Machinery, New York, NY, USA, 201–208. https://doi.org/10.1145/1133265.1133306

[27] Krzysztof Z. Gajos, Katherine Everitt, Desney S. Tan, Mary Czerwinski, and Daniel S. Weld. 2008. Predictability and Accuracy in Adaptive User Interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Florence, Italy) *(CHI '08).* Association for Computing Machinery, New York, NY, USA, 1271–1274. https://doi.org/10.1145/1357054.1357252

[28] Jens Grubert, Matthias Heinisch, Aaron Quigley, and Dieter Schmalstieg. 2015. 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* (Seoul, Republic of Korea) *(CHI '15).* Association for Computing Machinery, New York, NY, USA, 3933–3942. https://doi.org/10.1145/2702123.2702331

[29] Jens Grubert, Tobias Langlotz, Stefanie Zollmann, and Holger Regenbrecht. 2017. Towards Pervasive Augmented Reality: Context-Awareness in Augmented Reality. *IEEE Transactions on Visualization and Computer Graphics* 23, 6 (2017), 1706–1724. https://doi.org/10.1109/TVCG.2016.2543720

[30] Chris Harrison, Hrvoje Benko, and Andrew D. Wilson. 2011. OmniTouch: Wearable Multitouch Interaction Everywhere. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology* (Santa Barbara, California, USA) *(UIST '11).* Association for Computing Machinery, New York, NY, USA, 441–450. https://doi.org/10.1145/2047196.2047255

[31] Sandra G. Hart. 2006. Nasa-Task Load Index (NASA-TLX); 20 Years Later. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting* 50, 9 (2006), 904–908. https://doi.org/10.1177/154193120605000909

[32] Ke Huang, Chunhui Zhang, Xiaoxiao Ma, and Guanling Chen. 2012. Predicting Mobile Application Usage Using Contextual Information. In *Proceedings of the 2012 ACM Conference on Ubiquitous Computing* (Pittsburgh, Pennsylvania) *(UbiComp '12).* Association for Computing Machinery, New York, NY, USA, 1059–1065.

https://doi.org/10.1145/2370216.2370442

[33] Jonathan W. Kelly, Alec G. Ostrander, Alex F. Lim, Lucia A. Cherep, and Stephen B. Gilbert. 2020. Teleporting through virtual environments: Effects of path scale and environment scale on spatial updating. *IEEE Transactions on Visualization and Computer Graphics* 26, 5 (2020), 1841–1850. https://doi.org/10.1109/TVCG.2020.2973051

[34] Inyeop Kim, Hwarang Goh, Nematjon Narziev, Youngtae Noh, and Uichin Lee. 2020. Understanding User Contexts and Coping Strategies for Context-Aware Phone Distraction Management System Design. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 4, 4, Article 134 (dec 2020), 33 pages. https://doi.org/10.1145/3432213

[35] Pascal Knierim, Paweł W. Woźniak, Yomna Abdelrahman, and Albrecht Schmidt. 2019. Exploring the Potential of Augmented Reality in Domestic Environments. In *Proceedings of the 21st International Conference on Human-Computer Interaction with Mobile Devices and Services* (Taipei, Taiwan) *(MobileHCI '19)*. Association for Computing Machinery, New York, NY, USA, Article 31, 12 pages. https://doi.org/10.1145/3338286.3340142

[36] Ben Lafreniere, Tanya R. Jonker, Stephanie Santosa, Mark Parent, Michael Glueck, Tovi Grossman, Hrvoje Benko, and Daniel Wigdor. 2021. *False Positives vs. False Negatives: The Effects of Recovery Time and Cognitive Costs on Input Error Preference.* Association for Computing Machinery, New York, NY, USA, 54–68. https://doi.org/10.1145/3472749.3474735

[37] Wallace S. Lages and Doug A. Bowman. 2019. Walking with Adaptive Augmented Reality Workspaces: Design and Usage Patterns. In *Proceedings of the 24th International Conference on Intelligent User Interfaces* (Marina del Ray, California) *(IUI '19)*. Association for Computing Machinery, New York, NY, USA, 356–366. https://doi.org/10.1145/3301275.3302278

[38] Cha Lee, Scott Bonebrake, Doug A. Bowman, and Tobias Höllerer. 2010. The role of latency in the validity of AR simulation. In *2010 IEEE Virtual Reality Conference (VR)*. 11–18. https://doi.org/10.1109/VR.2010.5444820

[39] Yuan Li, Feiyu Lu, Wallace S Lages, and Doug Bowman. 2019. Gaze Direction Visualization Techniques for Collaborative Wide-Area Model-Free Augmented Reality. In *Symposium on Spatial User Interaction* (New Orleans, LA, USA) *(SUI '19)*. Association for Computing Machinery, New York, NY, USA, Article 11, 11 pages. https://doi.org/10.1145/3357251.3357583

[40] David Lindlbauer, Anna Maria Feit, and Otmar Hilliges. 2019. Context-Aware Online Adaptation of Mixed Reality Interfaces. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology* (New Orleans, LA, USA) *(UIST '19)*. Association for Computing Machinery, New York, NY, USA, 147–160. https://doi.org/10.1145/3332165.3347949

[41] Feiyu Lu and Doug A. Bowman. 2021. Evaluating the Potential of Glanceable AR Interfaces for Authentic Everyday Uses. In *2021 IEEE Virtual Reality and 3D User Interfaces (VR)*. 768–777. https://doi.org/10.1109/VR50410.2021.00104

[42] Feiyu Lu, Shakiba Davari, Lee Lisle, Yuan Li, and Doug A. Bowman. 2020. Glanceable AR: Evaluating Information Access Methods for Head-Worn Augmented Reality. In *2020 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*. 930–939. https://doi.org/10.1109/VR46266.2020.00113

[43] K. Makita, M. Kanbara, and N. Yokoya. 2009. View management of annotations for wearable augmented reality. In *2009 IEEE International Conference on Multimedia and Expo*. 982–985. https://doi.org/10.1109/ICME.2009.5202661

[44] Jakob Nielsen. 1994. Usability Inspection Methods. In *Conference Companion on Human Factors in Computing Systems* (Boston, Massachusetts, USA) *(CHI '94)*. Association for Computing Machinery, New York, NY, USA, 413–414. https://doi.org/10.1145/259963.260531

[45] Don Norman. 2009. *The design of future things.* Basic books.

[46] Chen Qu, Liu Yang, W. Bruce Croft, Yongfeng Zhang, Johanne R. Trippas, and Minghui Qiu. 2019. User Intent Prediction in Information-Seeking Conversations. In *Proceedings of the 2019 Conference on Human Information Interaction and Retrieval* (Glasgow, Scotland UK) *(CHIIR '19)*. Association for Computing Machinery, New York, NY, USA, 25–33. https://doi.org/10.1145/3295750.3298924

[47] Philip Quinn and Andy Cockburn. 2020. Loss Aversion and Preferences in Interaction. *Human–Computer Interaction* 35, 2 (2020), 143–190. https://doi.org/10.1080/07370024.2018.1433040

[48] Philip Quinn and Shumin Zhai. 2016. A Cost-Benefit Study of Text Entry Suggestion Interaction. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (San Jose, California, USA) *(CHI '16)*. Association for Computing Machinery, New York, NY, USA, 83–88. https://doi.org/10.1145/2858036.2858305

[49] Robert Rosenthal, Harris Cooper, Larry Hedges, et al. 1994. Parametric measures of effect size. *The handbook of research synthesis* 621, 2 (1994), 231–244.

[50] Edward Rosten, Gerhard Reitmayr, and Tom Drummond. 2005. Real-time video annotations for augmented reality. In *International Symposium on Visual Computing*. Springer, 294–302. https://doi.org/10.1007/11595755_36

[51] Quentin Roy, Futian Zhang, and Daniel Vogel. 2019. Automation Accuracy Is Good, but High Controllability May Be Better. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland Uk) *(CHI '19)*. Association for Computing Machinery, New York, NY, USA, 1–8. https://doi.org/10.1145/3290605.3300750

[52] SAE. 2021. *Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles.* Retrieved February 23, 2022 from https://www.sae.org/standards/content/j3016_202104/

[53] Jeff Sauro and Joseph S. Dumas. 2009. Comparison of Three One-Question, Post-Task Usability Questionnaires. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Boston, MA, USA) *(CHI '09)*. Association for Computing Machinery, New York, NY, USA, 1599–1608. https://doi.org/10.1145/1518701.1518946

[54] Dennis Schleicher, Peter Jones, and Oksana Kachur. 2010. Bodystorming as Embodied Designing. *Interactions* 17, 6 (nov 2010), 47–51. https://doi.org/10.1145/1865245.1865256

[55] Choonsung Shin, Jin-Hyuk Hong, and Anind K. Dey. 2012. Understanding and Prediction of Mobile Application Usage for Smart Phones. In *Proceedings of the 2012 ACM Conference on Ubiquitous Computing* (Pittsburgh, Pennsylvania) *(UbiComp '12)*. Association for Computing Machinery, New York, NY, USA, 173–182. https://doi.org/10.1145/2370216.2370243

[56] B. Shneiderman. 1993. Beyond intelligent machines: just do it. *IEEE Software* 10, 1 (1993), 100–103. https://doi.org/10.1109/52.207235

[57] Timothy Sohn, Kevin A. Li, William G. Griswold, and James D. Hollan. 2008. A Diary Study of Mobile Information Needs. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Florence, Italy) *(CHI '08)*. Association for Computing Machinery, New York, NY, USA, 433–442. https://doi.org/10.1145/1357054.1357125

[58] Adam Tapal, Ela Oren, Reuven Dar, and Baruch Eitam. 2017. The Sense of Agency Scale: A Measure of Consciously Perceived Control over One's Mind, Body, and the Immediate Environment. *Frontiers in Psychology* 8 (2017). https://doi.org/10.3389/fpsyg.2017.01552

[59] Jaime Teevan, Susan T. Dumais, and Daniel J. Liebling. 2008. To Personalize or Not to Personalize: Modeling Queries with Variation in User Intent. In *Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (Singapore, Singapore) *(SIGIR '08)*. Association for Computing Machinery, New York, NY, USA, 163–170. https://doi.org/10.1145/1390334.1390364

[60] April Tyack, Peta Wyeth, and Daniel Johnson. 2020. *Restorative Play: Videogames Improve Player Wellbeing After a Need-Frustrating Event.* Association for Computing Machinery, New York, NY, USA, 1–15. https://doi.org/10.1145/3313831.3376332

[61] Leena Ventä-Olkkonen, Jonna Häkkilä, and Kaisa Väänänen-Vainio-Mattila. 2014. Exploring the Augmented Home Window: User Perceptions of the Concept. In *Proceedings of the 13th International Conference on Mobile and Ubiquitous Multimedia* (Melbourne, Victoria, Australia) *(MUM '14)*. Association for Computing Machinery, New York, NY, USA, 190–198. https://doi.org/10.1145/2677972.2677994

[62] Lars Vising. 2008. Mobile Widget Architecture.

[63] Jacob O. Wobbrock, Leah Findlater, Darren Gergle, and James J. Higgins. 2011. The Aligned Rank Transform for Nonparametric Factorial Analyses Using Only Anova Procedures. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Vancouver, BC, Canada) *(CHI '11)*. Association for Computing Machinery, New York, NY, USA, 143–146. https://doi.org/10.1145/1978942.1978963

[64] Wolfgang Woerndl, Christian Schueller, and Rolf Wojtech. 2007. A Hybrid Recommender System for Context-aware Recommendations of Mobile Applications. In *2007 IEEE 23rd International Conference on Data Engineering Workshop*. 871–878. https://doi.org/10.1109/ICDEW.2007.4401078

[65] Jason Wu, Chris Harrison, Jeffrey P. Bigham, and Gierad Laput. 2020. *Automated Class Discovery and One-Shot Interactions for Acoustic Activity Recognition.* Association for Computing Machinery, New York, NY, USA, 1–14. https://doi.org/10.1145/3313831.3376875

[66] Congying Xia, Chenwei Zhang, Xiaohui Yan, Yi Chang, and Philip S Yu. 2018. Zero-shot user intent detection via capsule neural networks. *arXiv preprint arXiv:1809.00385* (2018). https://arxiv.org/abs/1809.00385

[67] Xusheng Xiao, Xiaoyin Wang, Zhihao Cao, Hanlin Wang, and Peng Gao. 2019. IconIntent: Automatic Identification of Sensitive UI Widgets Based on Icon Classification for Android Apps. In *2019 IEEE/ACM 41st International Conference on Software Engineering (ICSE)*. 257–268. https://doi.org/10.1109/ICSE.2019.00041

[68] Ye Xu, Mu Lin, Hong Lu, Giuseppe Cardone, Nicholas Lane, Zhenyu Chen, Andrew Campbell, and Tanzeem Choudhury. 2013. Preference, Context and Communities: A Multi-Faceted Approach to Predicting Smartphone App Usage Patterns. In *Proceedings of the 2013 International Symposium on Wearable Computers* (Zurich, Switzerland) *(ISWC '13)*. Association for Computing Machinery, New York, NY, USA, 69–76. https://doi.org/10.1145/2493988.2494333

[69] Difeng Yu, Qiushi Zhou, Benjamin Tag, Tilman Dingler, Eduardo Velloso, and Jorge Goncalves. 2020. Engaging Participants during Selection Studies in Virtual Reality. In *2020 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*. 500–509. https://doi.org/10.1109/VR46266.2020.00071

[70] Fabio Massimo Zanzotto. 2019. Human-in-the-loop Artificial Intelligence. *Journal of Artificial Intelligence Research* 64 (2019), 243–252. https://doi.org/10.1613/jair.1.11345

[71] Le Zhang, Qing-Xing Qu, Wen-Yu Chao, and Vincent G. Duffy. 2020. Investigating the Combination of Adaptive UIs and Adaptable UIs for Improving Usability and

User Performance of Complex UIs. *International Journal of Human–Computer Interaction* 36, 1 (2020), 82–94. https://doi.org/10.1080/10447318.2019.1606975

[72] Xun Zou, Wangsheng Zhang, Shijian Li, and Gang Pan. 2013. Prophet: What App You Wish to Use Next. In *Proceedings of the 2013 ACM Conference on Pervasive and Ubiquitous Computing Adjunct Publication* (Zurich, Switzerland) *(UbiComp '13 Adjunct).* Association for Computing Machinery, New York, NY, USA, 167–170. https://doi.org/10.1145/2494091.2494146