Designing a Multitasking Interface for Object-aware AR applications

Brandon Huynh[*]
University of California, Santa Barbara

Jason Orlosky[†]
Osaka University

Tobias Höllerer[‡]
University of California, Santa Barbara

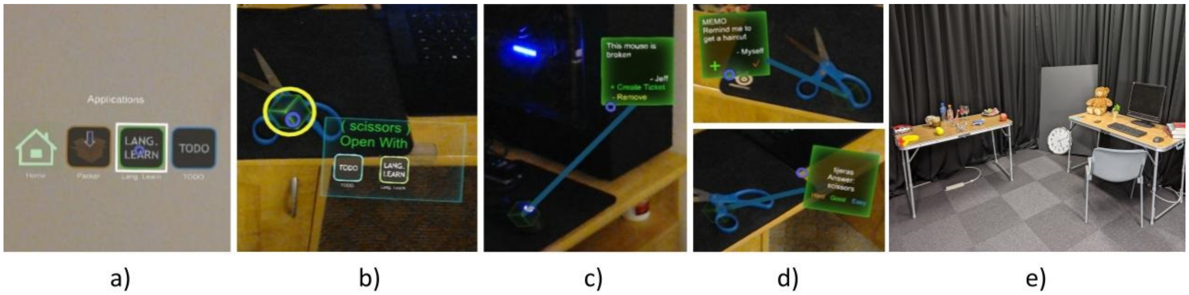

Figure 1: The multitasking interfaces we designed, including a) Static App Switcher and b) In-situ App Switcher, along with 3 prototype AR applications including c) Packer: a workplace assistance application, d) ToDo: for managing day-to-day reminders (top), and Lang Learn: for foreign language flash cards (bottom), and e) the testing environment.

Abstract

Many researchers and industry professionals believe Augmented Reality (AR) to be the next step in personal computing. However, the idea of an always-on context-aware AR device presents new and unique challenges to the way users organize multiple streams of information. What does multitasking look like and when should applications be tied to specific elements in the environment? In this exploratory study, we look at one such element: physical objects, and explore an object-centric approach to multitasking in AR. We developed 3 prototype applications that operate on a subset of objects in a simulated test environment. We performed a pilot study of our multitasking solution with a novice user, domain expert, and system expert to develop insights into the future of AR application design.

Index Terms: Human-centered computing—Mixed / augmented reality; Human-centered computing—User interface design

1 Introduction

The most popular commercial headsets of today, such as the Microsoft HoloLens and the Magic Leap One, use an amalgam of traditional window-based UI design, and smartphone style single-focus application design. Apps are launched through a home menu and placed in a physical location. The user can then interact with the app, either in the form of a 2D application window, or as an immersive application that can render anywhere over the existing environment. Multitasking occurs easily among 2D windows by placing them in non-overlapping locations. Immersive applications on the other hand require the user to entirely exit one application to start the next one. As AR applications grow more complex and context-aware, this design becomes untenable.

Some recent works have examined how to manage multiple applications in context-aware settings. For example, Lages et al. [3] adapted the physical layout of windowed applications to the wall structure of rooms as the user walked around. In this work, we consider another case, applications that depend on or react to physical objects. In this scenario, switching between applications through existing approaches would be difficult as users cannot be expected to know all the possible objects that are relevant to a particular application. We sought to design a way to signal these context dependencies within a multitasking interface.

We designed an object-aware user interface which recommends available applications for specific objects. The interface is presented as co-located with the physical object to reduce multitasking friction. We compared this work to the aforementioned home menu approach that is used by existing HMDs. We conducted an initial evaluation through a pilot study with 3 participants of different levels of expertise: novice user, domain expert, and system expert. After analyzing their comments and usage patterns, we identify promising research topics to stimulate further discussion.

2 Hardware and Software Apparatus

We used the Microsoft HoloLens as our headset of choice, with input through an Xbox one controller for experimental flexibility. The clicker or air-tapping would be sufficient as well. The object detection procedure is based on a client-server model that sends frames to an external server for the GPU-intensive object detection algorithm, similar to the method used by Huynh et al. [2]. Of note is that the object recognition is intentionally not simulated, as we wanted to see how these systems might function in the wild. The rate and order of detection varies based on how the user moves and looks around the environment.

2.1 Multitasking Interfaces

We designed 3 presentation conditions for our multitasking interface. **Static App Switcher** is the home screen app-drawer approach where users select another application by first opening a static menu that appears directly in front of the user's head and choosing from a list of all installed applications. This is designed to mimic the UI found in current HMDs. **In-situ App Switcher** is our novel situated and object-aware approach, where users select from a list of compatible applications for a specific object through a menu which is placed directly next to each object. **Combined Selection** allows the user to activate either type of menu and use them interchangeably. These conditions can be seen in Figure 1a and 1b.

2.2 Prototype Applications

We designed and implemented 3 prototype AR applications that operate on real-world objects, as shown in Figure 1c and 1d. **ToDo** uses situated post-it notes to assist in the creation of a reminder

[*] e-mail: bhuynh@cs.ucsb.edu
[†] e-mail: orlosky@lab.ime.cmc.osaka-u.ac.jp
[‡] e-mail: holl@cs.ucsb.edu

978-1-7281-7675-8/20/$31.00 ©2020 IEEE
DOI 10.1109/ISMAR-Adjunct51615.2020.00025

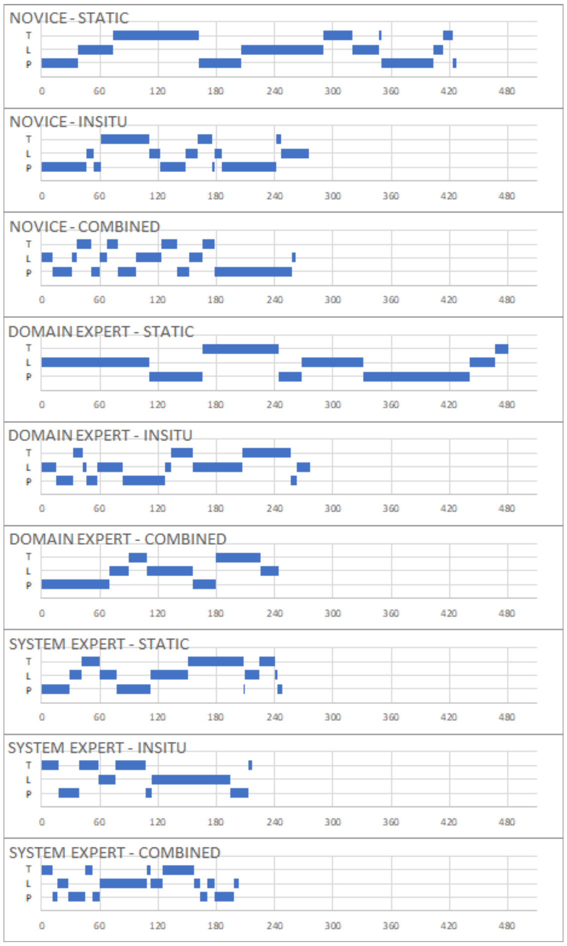

Figure 2: App usage in seconds for ToDo (T), Lang. Learn (L) and Packer (P).

list for everyday tasks, much like the work of Rekimoto et al. [4]. They are presented as being previously placed by friends and family. **Lang Learn** presents situated flash cards to assist in learning foreign language vocabulary words for encountered objects, inspired by a growing trend in AR for Education research [1]. **Packer** is an app to help workplace facility managers in locating defective equipment.

3 Novice vs. Experts Pilot Study

For the pilot study, we assigned 12 objects per application. Each app had 3 unique objects, 3 overlapping with another application, and 3 in common with all applications, for a total of 21 objects. The study environment can be seen in Figure 1e. Each app presented a small subtask per associated objects. The subtask consisted of a multiple choice prompt related to the scope of the app. For instance, in the ToDo app, a prompt would appear asking if the user wants to add the object's memo to their reminder list. The primary task was to get through all 36 subtasks as quickly as possible with a given presentation condition. Participants completed the primary task 3 times, first with Static, then In-Situ, and finally Combined conditions. The study was conducted with 3 users. The novice user had no previous AR experience. The domain expert had plenty of experience with AR, but not our system. The system expert is both an AR expert and had tested our system extensively.

3.1 Results

Timelines of app usage is shown in Figure 2, allowing us to observe usage patterns. The first thing we see is that participants took the longest in the Static condition, though this may be due to order effects. We also noticed that app sessions were longer during the Static condition, averaging 60.1 seconds vs. 21.3 and 34.8 seconds in the In-Situ and Combined conditions respectively. In the Combined condition, participants were free to use either the static or in-situ switcher, but the participants chose in-situ switcher the majority (73%) of the time. In fact, the system and domain expert only used the static switcher once, and the novice only 4 times.

We also conducted short semi-structured interviews with topics grouped into 3 categories. We include comments from the Domain Expert (DE) and Novice User (NU).

Status Visibility: *Was it easy to understand which objects had augmentations remaining?* - DE had an easier time understanding how many subtasks were left with in-situ and combined. NU felt that only with in-situ could she "tell if there was anything left."

Ease of Use: *Which modality felt more natural to you?* - DE said in-situ. NU also choose the same, mentioning it felt natural to fidget with it when they weren't sure what else to do. *Did you feel like you did a lot of unnecessary switching in any modality?* - DE responded "Not really.". NU responded "maybe in the first [static] one because I was going through the apps in order", referencing participant resorting to scanning behaviour to find last two objects. *Did one feel faster than the others* - DE responded with in-situ and combined. NU responded with combined.

Overall Preference: *Which interface did you prefer the most and why?* - DU preferred combined as it was the most flexible. He also felt that the in-situ would work just fine, as he used the context menu the majority of the time. He mentioned "It's good because when I see the object I was kind of remind [sic] what I have left [to do]." NU preferred combined as well for similar reasons. Mentioned using in-situ menu most of the time "to check what was left if I couldn't find anything else to do."

4 Discussion and Conclusion

We explored the problem of multitasking among object-dependent AR applications and designed a multitasking approach that contextualizes application switching by co-locating the interface with the physical objects. We conducted a pilot study to evaluate our interface, comparing opinions and usage patterns from the perspective of a novice user, domain expert, and system expert.

Two interesting observations emerged from our results. First, participants completed the task faster when using the in-situ switcher. They also felt faster in the interviews. This is encouraging and suggests that our approach is easier to use and reduces friction. Second, participants had an easier time understanding the current state of augmentations with in-situ. This may be due to explicitly visualizing object-application connections, or due to faster or more convenient switching afforded by the interface.

This work was partially supported by ONR grant N00014-19-1-2553, ONR grant N62909-18-1-2036 and NSF grant IIS-1845587.

References

[1] F. Draxler, A. Labrie, A. Schmidt, and L. L. Chuang. Augmented reality to enable users in learning case grammar from their real-world interactions. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, CHI '20, p. 1–12. Association for Computing Machinery, New York, NY, USA, 2020. doi: 10.1145/3313831.3376537

[2] B. Huynh, J. Orlosky, and T. Höllerer. In-situ labeling for augmented reality language learning. In *2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 1606–1611. IEEE, 2019.

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

[4] J. Rekimoto and Y. Ayatsuka. Cybercode: designing augmented reality environments with visual tags. In *Proceedings of DARE 2000 on Designing augmented reality environments*, pp. 1–10, 2000.