University of California
Santa Barbara

Design and Evaluation of Pervasive Augmented Reality Systems

A dissertation submitted in partial satisfaction
of the requirements for the degree

Doctor of Philosophy
in
Computer Science

by

Brandon Huynh

Committee in charge:

Professor Tobias Höllerer, Chair
Professor Matthew Turk
Professor Misha Sra
Professor Jennifer Jacobs
Professor Jason Orlosky, Augusta University

December 2022

The Dissertation of Brandon Huynh is approved.

Professor Matthew Turk

Professor Misha Sra

Professor Jennifer Jacobs

Professor Jason Orlosky, Augusta University

Professor Tobias Höllerer, Committee Chair

September 2022

Design and Evaluation of Pervasive Augmented Reality Systems

by

Brandon Huynh

Dedicated to Mom and Dad.

Acknowledgements

Thank you to my advisor Tobias Höllerer, for your guidance and mentorship. You pushed me to make hard choices and taught me to believe in my ideas. You gave me the room to fail, and were always there to pick me back up. You were kind and you were patient with me. You didn't just teach me to be a better researcher, you taught me how to be a better person.

Thank you to Jason Orlosky for your guidance and mentorship. Your enthusiasm for research is infectious. You always saw the upside to every project, and helped me remember why I started a PhD in the first place. You kept me grounded, taught me to write more, and think pragmatically about my work.

Thank you to Matthew Turk, for teaching me how to read papers effectively and think critically about research. Thank you to my other committee members, Jennifer Jacobs and Misha Sra, for lending your ears and for your valuable insights.

Thank you to my colleagues and collaborators: Tawny Lim, James Schaffer, Yun Suk Chang, Benjamin Nuernberger, Steffen Gauglitz, Donghao Ren, John O'Donovan, Jonathan Downey, Dorothy Chun, Peter (Zhe Fu), Yi Ding, Ehsan Sayyad, You-Jin Kim, CY (Chengyuan Xu), Aiwen Xu, Tom Bullock, Barry Giesbrecht, Pushkar Shukla, Radha Kumaran, Alex Rich, Noah Stier, Photchara Ratsamee, Abby Wysopal, Vivian Ross, and others I may have forgotten.

Thank you to the wonderful community of researchers and friends that I have made at UCSB's Four Eyes Lab and Osaka University's Takemura Lab.

Thank you to my friends, Matt, Jeff, and Christine, for always being there. Thank you to Krystal for always listening to my rants and commiserating with me. We finally did it.

Thank you to my lovely partner Kelly for always supporting me all these years. I couldn't have done it without you.

Curriculum Vitæ
Brandon Huynh

Education

2022	Ph.D. in Computer Science, University of California, Santa Barbara
2020	M.S. in Computer Science, University of California, Santa Barbara
2015	B.S. in Computer Science, University of California, Riverside

Work Experience

Graduate Student Researcher, Four Eyes Lab
UC Santa Barbara, Santa Barbara, USA 2016 - 2022

Specially Appointed Researcher, Cybermedia Center
Osaka University, Osaka, Japan 2018 - 2020

Research Intern, Vuforia Computer Vision Team
PTC Inc., Vienna, Austria Summer 2017

Research Intern, Mobile Vision Team
Google, Los Angeles, USA Summer 2016

Software Engineer
Caugnate, Santa Barbara, USA 2016

Software Engineering Intern, Chrome Protector Team
Google, Montreal, Canada Summer 2015

Software Engineering Intern, Chromecast Team
Google, Mountain View, USA Summer 2014

Undergraduate Research Assistant, Riverside Graphics Lab
UC Riverside, Riverside, USA 2012 - 2015

Mentoring

2022	Abby Wysopal, M.S. at UC Santa Barbara
2022	Vivian Ross, M.S. at UC Santa Barbara
2019	Sofia Onyshko, M.S. at Osaka University
2016	Xijun Wang, Ph.D. at Northwestern University

Awards

2018	Finalist, Qualcomm Innovation Fellowship
2017	Honorable Mention, NSF Graduate Student Research Fellowship

Publications

1. **Brandon Huynh**, Abby Wysopal, Vivian Ross, Jason Orlosky, Tobias Höllerer. "Layerable Apps: Comparing Concurrent and Exclusive Display of Augmented Reality Applications." In 2022 IEEE International Symposium on Mixed and Augmented Reality (ISMAR), pp. TBD
2. **Brandon Huynh**, Jason Orlosky, Tobias Höllerer. "Designing a Multitasking Interface for Object-aware AR applications." In 2020 IEEE International Symposium on Mixed and Augmented Reality Adjunct (ISMAR-Adjunct), pp. 39-40.
3. Jason Orlosky, **Brandon Huynh**, Tobias Höllerer. "Using eye tracked virtual reality to classify understanding of vocabulary in recall tasks." In 2019 IEEE International Conference on Artificial Intelligence and Virtual Reality (AIVR), pp. 66-73.
4. Yi Ding, **Brandon Huynh**, Aiwen Xu, Tom Bullock, Hubert Cecotti, Barry Giesbrecht, Tobias Höllerer. "Multimodal classification of EEG during physical activity." In 2019 International Conference on Multimodal Interaction (ICMI), pp. 185-194.
5. **Brandon Huynh**, Adam Ibrahim, Yun-Suk Chang, Tobias Höllerer, John O'Donovan. "User perception of situated product recommendations in augmented reality." International Journal of Semantic Computing 13, no. 03 (2019), pp. 289-310.
6. **Brandon Huynh**, Jason Orlosky, Tobias Höllerer. "In-situ labeling for augmented reality language learning." In 2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR), pp. 1606-1611.
7. **Brandon Huynh**, Jason Orlosky, Tobias Höllerer. "Semantic labeling and object registration for augmented reality language learning." In 2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR), pp. 986-987.
8. **Brandon Huynh**, Adam Ibrahim, Yun-Suk Chang, Tobias Höllerer, John O'Donovan. "A study of situated product recommendations in augmented reality." In 2018 IEEE International Conference on Artificial Intelligence and Virtual Reality (AIVR), pp. 35-43.
9. Adam Ibrahim, **Brandon Huynh**, Tobias Höllerer, John O'Donovan. "Arbis pictus: A study of vocabulary learning with augmented reality." IEEE transactions on visualization and computer graphics 24, no. 11 (2018), pp. 2867-2874.
10. James Schaffer, **Brandon Huynh**, John O'Donovan, Tobias Höllerer, Yinglong Xia and Sabrina Lin. "An analysis of student behavior in two massive open online courses." In 2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM), pp. 380-385.

Abstract

Design and Evaluation of Pervasive Augmented Reality Systems

by

Brandon Huynh

Augmented Reality (AR) is an interactive technology that delivers rich and immersive computer-generated perceptual information overlaid onto the real world. AR technology continues to improve over time, potentially enabling an ”always-on” AR future, where wearable AR devices are as comfortable to wear as glasses and have an all-day battery life. This concept has been termed ”Pervasive Augmented Reality”, in which AR is the predominant form of personal computing, instantly accessible, constantly providing information and available to assist in everyday tasks.

Pervasive AR represents a shift in how, when, and where we use computers, but there are few guidelines for how to effectively create these types of systems. Current research focuses on single-purpose use cases, designed for specific tasks and environments, such as navigation or maintenance. They rarely consider AR as a mobile experience that can be taken anywhere, where computing happens continuously and persistently. This dissertation addresses that insufficiency by analyzing the design and evaluation of AR artifacts for 3 use cases: AR Recommender Systems, AR Language Learning, and AR Multitasking. We conduct our investigation in 3 parts. 1) What benefits do AR applications provide over non-AR based systems? 2) What additional inputs and signals can AR applications benefit from? and 3) How do we evolve current AR systems towards a pervasive AR future? Our work makes contributions through empirical user studies, prototype systems, and the development of new interaction techniques, revealing insights into the tools and techniques needed for developing pervasive AR, as well as the opportunities and new possibilities it enables over existing technologies.

Contents

Chapter 1

Introduction

Augmented Reality enhances the user's experience of the real world through computer generated perceptual information. Research on Augmented Reality (AR) traces its roots back to Ivan Sutherland's demonstration of a head-mounted display in 1968, though the idea of adding information to the physical world to annotate it likely goes back even further. Historically, research has focused on enabling technologies for AR devices, such as enhancements in displays, camera sensors, and 3D registration algorithms. That research culminated in the widespread availability of AR camera applications today, turning AR from niche concept to widely recognized computing paradigm.

Large technology companies are now investing billions of dollars to develop better head-mounted AR displays. Products like the Microsoft HoloLens can map the environment in real time with increasing accuracy, using that information to deliver virtual graphics that exhibit realistic dynamic behavior, such as occluding virtual objects behind real objects. Trends in the development of AR hardware as well as adjacent technologies suggest that future devices may very well be "always-on", similar to and even more so than smartphones.

Comfort and Display Size. Current headsets are heavy and difficult to wear for long periods of time. However, there are already a number of industrial research prototypes with much smaller, more lightweight displays, including the North Focals, Nreal Glasses, and Meta's Project Aria, whose form factor more closely resembles a pair of eyeglasses. We believe it is reasonable to expect future displays to be not much larger than standard prescription eyewear, especially if computation is offloaded to an external device such as a smartphone.

Battery Life. Today's standalone HMDs have a battery life of about three hours, which is not ideal for everyday use. Tethered devices such as the Magic Leap AR headset offer one potential solution, as they can improve on battery life by offloading battery storage and computation to another more thermally efficient device. Increased power efficiency from smaller semiconductor manufacturing processes can be expected from a variant of Moore's law. While the HoloLens uses a Qualcomm system-on-chip (SoC) fabricated with a 10 nanometer (nm) process, 5nm chips are already in production for other mobile devices, such as Apple's iPad, with 3nm and even 2nm processes estimated to arrive by 2025. Finally, battery capacity improvements from other technologies, like electric cars, might eventually trickle down to these devices.

Connectivity. Many of the current AR devices use the same SoC components typically found in modern smartphones. Thus it is reasonable to expect that connectivity features, such as 5G networking, will be easy to integrate into future devices. This would provide access to a variety of services that would be difficult to achieve on the fly, or with limited computation power. Detailed models of public environments could be downloaded on the fly to improve tracking. Sensor data could be streamed to machine learning services to semantically analyze the user's context.

These trends suggest future devices will be lighter and more comfortable, last longer, and be powered by intelligent algorithms. The improved ergonomics will encourage a continuous and near omnipresent AR experience, shifting from the sporadic single-purpose usage of today. The possibility of a computer system having a constant role in our lives presents new design challenges, and more and more researchers have shifted from enabling technologies to enabling interfaces and experiences. In a 2018 survey of papers published at the International Symposium of Mixed and Augmented Reality (ISMAR), Kim et al. distilled research trends from the last 10 years and proposed research directions for the next decade [1]. They identified a trend towards increasing context-awareness and semantic understanding and highlighted challenges in human factors research. A new vision is emerging of Augmented Reality as a personal computing medium, taking the act of computing out of individual devices and embedding them into the physical world. The next challenge is not in determining whether these displays are feasible, but how we will interact with them.

Grubert et al. provided a name for this concept, termed “Pervasive Augmented Reality” [2]. Other similar terms include “Always-on Augmented Reality” or “Context-aware Augmented Reality”. Pervasive AR represents a dramatic shift in the way we use computers. Never has the operation of a computer system been so closely aligned with our own perception of the world. AR systems can directly augment the information received by our senses. It is imperative that we understand the right and wrong ways to develop these systems. This dissertation focuses on generating new knowledge in the design and evaluation of pervasive AR systems, recognizing that these systems need to be developed ethically and responsibly.

1.1 Motivation

Augmented Reality research has expanded significantly in the past two decades, with the increased availability of affordable AR displays. Most of this work has focused on addressing technical and user experience challenges, such as spatial mapping, view management, or interaction techniques [1, 3]. There hasn’t been much work focusing on the pervasive qualities of the medium, namely the combination of mobility, presence, and context sensitivity [4, 5]. AR headsets can potentially be operated on the move, anywhere in the world, unlike desktop PCs. They can immerse your senses, unlike smartphones. And they situate their content directly onto your physical environment, unlike VR headsets. Together, these features produce a distinct computing medium, in search of a new operating paradigm.

The construction of systems in this paradigm is an open research challenge, as they touch upon many domains that have not been traditionally studied in the AR literature. Take, for instance, the role and impact of human perception and cognition. Work has been done on core aspects of visual perception, such as increasing realism in AR graphics [6–8], but compared to the desktop space [9], we know very little about the other cognitive processes invoked when using AR. Another example is the presentation and visualization of AR content. Most AR applications are built to be used in a specific environment and can be optimized through trial and error for that environment. But once we start to move around in the world, the placement and integration of AR content becomes much more complex. Researchers have only recently begun to explore solutions to this problem [10, 11].

This work aims to address these shortcomings and inform the design of future AR experiences through the investigation of three key research questions:

- What benefits do AR applications provide over non-AR based systems?
- What additional inputs and signals can AR applications benefit from?
- How do we evolve current AR systems towards a pervasive AR future?

I chose these questions because I believe the answers will help to drive increased adoption of Augmented Reality. In the following sections, I provide further justifications for these modes of inquiry, and also provide a brief overview of the background literature that informed my choice of projects in service of these questions.

1.1.1 Benefits of AR

What benefits do AR applications provide over non-AR-based systems? The goal of this question is to help identify the value proposition for pervasive AR. Despite significant progress, it is still unclear what the distinct benefits and advantages of using AR systems are, and it is difficult to convey why they should be more widely adopted to a general audience. For AR to become a mainstream computing platform, we need to identify what value it can provide for the average user, and what distinct qualities developers should focus their applications around. By answering this question, we can provide guidelines for developers to assess whether their applications will be effectively persuade users away from existing technologies.

There has been work done to identify benefits in the context of specific tasks, such as in the deployment of AR for maintenance and repair tasks. Henderson and Feiner, for example, developed a head-worn AR prototype for mechanics to use when repairing armored personnel carriers [12]. Their study found that mechanics were able to identify task locations more quickly using the AR system, compared to the same information presented on an LCD screen. In a related study, the same authors demonstrated an AR prototype to provide assistance during procedural maintenance tasks [13]. Again comparing to an LCD display, they found AR to be faster during the psychomotor phases of these tasks. These works demonstrated the value of AR in its ability to assist users in accomplishing spatial tasks.

Researchers have also explored the use of AR for assembly tasks, particularly in factory settings. One of the earliest instances was at Boeing research, where Caudell and Mizell explored the use of an AR headset for assembling wire bundles in an aircraft [14]. This was later tested in the field by Curtis et al. in 1999 [15]. Later works directly compared the use of AR against other forms of assembly instructions, including paper manuals [16–18] and computer screens [16]. These works generally found AR to yield faster assembly times with fewer errors.

It is clear that AR can succeed when applied to tasks that have a high degree of spatial coordination. However, these types of tasks are niche and typically performed by those with additional prior training and education, rather than your everyday computer user. Additionally, due to the highly technical nature of the operations, there are few alternative methods besides text instructions to compare against, making their results difficult to generalize. In contrast, applications of AR to tasks in personal and domestic settings are less explored. In these settings, pervasive AR systems are more likely to compete with existing forms of consumer technology, such as smartphones or online services.

One example of a personal use case where AR has recently become popularized is shopping or e-commerce. For instance, smartphone applications such as those provided by IKEA and Houzz allow users to visualize 3D models of furniture in their living spaces to assist in purchasing decisions. Clothing companies like ZOZO and Nike are experimenting with AR apps to allow users to virtually try on clothing. However, the majority of research in this space has focused more on the benefits of using AR with regards to marketing and branding [19, 20]. While these benefits are certainly appealing to businesses, they don't provide many insights on the potential incentives for end users. Additionally, as these works focus on smartphone based AR, it is unclear how many of these effects would hold in head-worn AR.

Billinghurst et al. summarized decades of research in their 2015 Augmented Reality survey [4], and in it they also identified modern applications of AR in a variety of domains. Unsurprisingly, most of these domains were also highly technical and industry specific, such as Marketing, Medicine, and Architecture. However, one domain they focused on could be considered a personal use case: Education. While education technology is usually driven by external forces such as school boards and curriculum standards, it is also something private individuals choose to adopt. Many education technologies and products exist that are not adopted by schools but still used by consumers on a daily basis, such as spaced repetition applications [21] or gamified learning tools [22].

An early example of the use of AR in education is the MagicBook project [23], which used a physical book to facilitate transitions between the real world and virtual reality content. Later work on AR books have focused on the use of physical books with tracking markers that indicate AR content users can interact with, viewed through a secondary device such as a mobile computer or PC computer with a webcam [24, 25]. The results of these studies are encouraging as they demonstrated slight improvements to learning outcomes when using AR. However, they are also not using head-worn AR, so it is unclear whether factors like immersion or visual-spatial processing would have an effect.

Comparative research that identifies the benefits of AR has largely focused on highly specialized applications or non-spatial presentations of content. These are either too specific to be relevant to your average consumer, or they aren't taking into consideration other important qualities of future AR systems that are more mobile and spatially aware. To really understand what are the benefits of AR, especially when used pervasively, we need to be looking at consumer use cases, and comparing them to other consumer technologies. In chapters 2 and 3 of this dissertation, we extend the existing body of work through new studies focusing on the e-commerce and education scenarios. By using prototype applications deployed onto spatial mapping enabled AR headsets, the benefits we discover will be more applicable to pervasive AR. In both instances, we also implement the same algorithms and graphical content into an existing consumer technology. This allows us to evaluate their acceptability as replacements, as users are already familiar with the existing technology and could speculate on whether they preferred to use our AR versions instead.

1.1.2 Additional Inputs and Signals

What additional inputs and signals can AR applications benefit from? The field is still in the early stages of understanding the needs and demands of pervasive AR, and there is not an accepted standard for what technical capabilities should be incorporated into a pervasive AR system. By answering this question, we provide researchers, designers, and platform developers with evidence to help them assess where to direct their efforts. This is especially valuable for design considerations of large software libraries and platform APIs, which will ultimately influence the kinds of pervasive AR apps that get created.

The taxonomy laid out by Grubert et al. provided a broad overview of the concepts of pervasive AR, and highlighted additional work that needs to be done to make it a reality [2]. When summarizing the existing body of research, they discussed the surprising lack of works on performance optimization and energy use optimization (only [26–28]), as well as the lack of works utilizing user-centric context sources (e.g. current task, physical conditions, cognitive factors), as an input for managing AR content. Indeed, only two prior works featuring user context models reached the prototype state of investigation [29, 30] at the time of publication. Ultimately, the authors felt that existing approaches up till now were "isolated islands of topics" that were not mature enough for use in implementing pervasive AR, and that the biggest challenge ahead of us is to build AR interfaces that are actually multi-purpose and context-controlled.

Performance optimization is a notable oversight in the AR literature, considering that the desired goal of pervasive AR is to allow for long-term usage and increased context-awareness. Simultaneous Localization and Mapping (SLAM) algorithms, which AR systems utilize for spatial mapping, make increasing use of deep learning methods to improve accuracy at the cost of performance [31, 32]. State-of-the-art context-awareness algorithms now almost universally use large deep learning models for tasks such as object recognition [33] and natural language processing [34]. Unfortunately, model optimization and inference speed is usually an afterthought for most machine learning researchers. Minaee et al. surveyed image segmentation models using deep learning, and found only 7 models out of 38 that reported inference speed at all, and only two of which might be considered real-time (25 FPS or higher according to the authors) [35]. They concluded there was plenty of room for improvement in terms of both inference speed and memory efficiency, especially if we want to fit them into mobile devices. Given that researchers continue to focus on accuracy and create larger and more computationally expensive models, it is not clear how we can cram all these algorithms into a small AR form factor that is still comfortable to wear and has an all-day battery life.

There has been some work on improving the inference efficiency of deep learning models. Jacob et al. introduced the quantization of weights and activation functions as integers [36], reducing memory footprint by 4x and improving latency by up to 50% on some models. Other works have taken advantage of sparsity to improve performance, following the observation that traditional deep learning models are dense and over-parameterized while biological brains are typically hierarchical and sparse, reusing recurrent structures for different tasks [37]. This typically comes in the form of graph pruning, which reduces model size by removing infrequently used parts of the network, or parameter sharing, which exploits redundancy by combining and reusing similar weights in different clusters of the network [38–40]. Despite these efforts, we are still far away from using deep learning models in real-time and performance critical settings, let alone using multiple context-awareness models in AR simultaneously.

There is another possible solution to enable performant context-sensing, which is to offload computation to a separate device [41], or even to a remote server [28]. However, requiring an additional wearable computer, such as backpack computer, should only be a short-term solution, as it increases the form-factor of AR which would reduce comfort and likely drive away consumers. Remote servers may be the more desirable approach. With the ongoing deployment of 5G networking, it may soon be possible to send the large amounts of data needed for deep learning models through wireless networks [42].

Some or all of these challenges will need to be solved before we can develop full fledged pervasive AR systems. But simply waiting for the technologies to mature is not an ideal solution. Given that context-sensing will likely rely on large machine learning algorithms, if we apply them to augmented reality, we are potentially exposing tons of sensitive and private information about our daily lives in order to use a pervasive AR system. There needs to be a balance between the amount of data that users make available, and the goals that users want to accomplish. In order to understand what that balance should be, we need to start designing pervasive AR prototypes and evaluating them now, so that we are ready for these questions when the technology becomes more readily available for developers outside the AR space.

This dissertation makes new contributions to this body of work by looking at how to enable context-sensing inputs and signals with respect to one well defined use case: AR language learning. Specifically, we investigate the use of eye tracking and object recognition to modulate the presentation and spatial layout of learning content, in chapters 4 and 5 respectively. As part of these efforts, we develop and test the use of a remote server for offloading deep-learning-based context-awareness algorithms from the AR headset. By providing the perspective of feature development through a well defined use case, we can produce insights into the amount of contextual and sensor information necessary to enable certain features of a pervasive AR system. In doing so, we can create expectations around data privacy, allowing users to make decisions on the trade-off between desired privacy and AR tasks the user wants to accomplish. While primarily focusing on language learning, the lessons from these prototypes can also be applied to other pervasive AR systems. For instance, in chapter 6 of this dissertation, we apply techniques developed for AR content management using object recognition into our situated context menu for AR multitasking.

1.1.3 Evolution to Pervasive Systems

How do we evolve current AR systems towards a pervasive AR future? While the previous question addressed the development of technologies far into the future, we cannot forget that AR already exists in many forms right now. In the current tech landscape, system development often happens iteratively and incrementally [43], focusing on immediate goals instead of long-term requirements [44]. The direction of future systems is informed but what has already been built in the now, and what immediate things could be built next. By answering this question, we can identify and act on the incremental steps that can be taken in the short term to direct the course of AR system development towards the vision of pervasive AR.

Today's AR experiences are mostly static, sporadic, and single-purpose affairs, akin to Virtual Reality that simply happens to use your physical environment as a backdrop. These, along with hardware limitations such as small field of view, poor display fidelity, and short battery life, have made it difficult to market AR devices to consumers, with the Google Glass, HoloLens 1, and Magic Leap 1 being prime examples [45]. Newer devices, such as the HoloLens 2 and Magic Leap 2, have since shifted their target audience towards business and industry customers for that reason. In order for AR to be accepted as a personal computing paradigm, users will need significantly more varied use cases.

In the past, a main issue was likely a lack of understanding around what the best uses of augmented reality are. Researchers tried to take advantage of the immersive qualities of AR, using as much of the physical space as possible to create interesting and unique applications. Considerable work was focused on identifying novel applications such as maintenance [12], tourism [46], games [47], or medicine [48]. Relatively few works considered the use of AR in multi-purpose settings. Di Verdi et al. introduced ARWin and level-of-detail widgets [49, 50] for a desktop-like AR experience with multiple applications. Grubert et al. surveyed the development of AR browsers [51], multimedia platforms that display relevant information based on geolocation and points of interest. These applications built on the works of Feiner et al. [41], Höllerer et al. [52], and Kooper et al. [53], but the inclusion of many sources of information thanks to broader integration with the world wide web makes recent ARBrowser work more multi-purpose. Additionally, though these works supported multiple applications, they did not make use of automatic context detection. To move towards pervasive AR, we need UI mechanisms that handle use cases that are both multi-purpose and also context-aware.

In recent years, we have seen the mainstream acceptance of AR concepts through games such as Pokemon Go [54]. Suddenly, AR moves away from the realm of niche applications and laboratory prototypes and into the hands of millions of consumers. We can observe that AR can be used in a wide variety of tasks and situations, and it becomes apparent that we don't have many solutions for how to merge them into a single seamless interface. Coupled with the wider availability of AR capable devices, researchers are starting to consider how to actually use multiple applications simultaneously and concurrently. Recent concepts have emerged for display and interaction with multiple applications, especially in dynamic or mobile settings, such as Glanceable AR [11] and adaptive workspaces [10].

Today, the current bottleneck is the application models used by wearable AR platforms. As AR shares many similarities with VR when it comes to software and hardware infrastructure, it is no surprise that platform builders combine and conflate the two when making platform decisions. For example, the Microsoft HoloLens is the most popular series of AR headsets currently used by AR researchers. Yet it shares the same development tools and application lifecycle as Microsoft's VR headsets. This may be a practical business decision, but it severely limits what researchers can do with the platform, which in turn may dissuade them from developing and studying multi-purpose AR usage scenarios.

To resolve this bottleneck, we need to provide alternative application models for AR systems to adopt. This dissertation extends the existing body of work with new app model designs that support multitasking with multiple applications. Recent work like Glanceable AR has already looked at dynamic usage of applications in different situations, therefore we focus our efforts on other aspects that are important to pervasive AR. Namely, in chapter 6 we look at how to support multiple context-aware applications, and in chapter 7 we look at how to support multiple applications with different presentation styles, as well as how to support different levels of augmentation.

1.2 Scope and Objective

My work takes inspiration from and contributes to the emerging vision of pervasive augmented reality, a trend in AR research that emphasizes always-on accessibility, implicit interactions, context-awareness, and dynamic adaption of computer-generated content to the current social and physical situation. This dissertation can loosely be grouped into three sections, each structured as a response to the research questions laid out in previous sections.

In the first part of this dissertation, encompassing chapters two and three, I examine the potential benefits that AR systems can have in everyday tasks and situations. To accomplish this, I design prototypes for applications in two domains, E-Commerce and Foreign Language Learning in both AR and a contemporary medium (websites and mobile apps respectively), and conduct user studies, analyzing metrics such as user perception, trust, enjoyment, as well as task performance. The results of these studies contribute to a growing body of evidence demonstrating the beneficial effects and potential use cases for pervasive AR in domestic tasks.

The second part of this dissertation, encompassing chapters 4 and 5, focuses on investigating new inputs and signals that could be added to pervasive AR systems to improve their functionality. Continuing the language learning use case, I conducted investigations into what new capabilities would be necessary to realize a fully automated learner feedback loop. I identified two desirable capabilities, object detection and understanding classification, theorizing that the combination of these signals would enable a pervasive AR app to identify objects to use as semantic anchors for automatically generating educational content and automatically scaffold difficulty and progression for the learner based on their current understanding of said content. For object detection, I leverage recent advancements in deep learning algorithms from the Computer Vision field, and focus my efforts in the design of systems and technical infrastructure required to incorporate these algorithms into the low-power and real-time processing constraints of a head-mounted display. In the case of understanding classification, I explore the use of eye tracking signals captured via near-infrared light eye cameras in the similar but distinct space of Virtual Reality (VR) head-mounted displays. The research efforts in these chapters push the boundary of what is possible in terms of contextual and situational awareness in AR.

In the final part of this dissertation, encompassing chapters 6 and 7, I look at how to improve currently available AR systems by incorporating pervasive AR concepts in their design. This work ultimately came about after realizing that the current direction of the industry was trending towards monolithic all-encompassing applications owned entirely by a few stakeholders, creating a potential AR future that is monopolistic, exclusionary, and divided by the haves and have-nots. In order to counter these trends and offer a differing perspective, I conducted research into the usability of AR systems that extend contemporary AR with more democratic methods of program execution. In these chapters, I introduce two prototypes that separate app functionality and provide the user with more agency in determining what parts of their world is augmented and to what degree. I develop exemplar applications to be used within these prototypes, and conduct user studies to examine task performance, usability, and enjoyment while interacting and multitasking between said applications. Notably, I also examine differences between novice users and experienced users, insight which is valuable when considering the design of a system that might be incorporated into consumer AR products targeting mass markets. The results of this work provide an alternative and more inclusionary perspective into the why and how of AR applications design.

Pervasive Augmented Reality is an exciting prospect, with the potential to transform our relationship with computers. There are numerous challenges and questions that have yet to be answered within this vision. The contributions made in this dissertation improve our understanding of this space.

1.3 Potential Risks

With every generation of computing comes socio-technical disruption and unintended consequences. Pervasive AR is likely no different. There are significant pitfalls and potential risks to the deployment of these systems that must be addressed. In this section, I highlight some of these potential risks. Furthermore, I contextualize the contributions made in this dissertation with regards to these risks.

1.3.1 Deskilling and Reality Distortion

In 2007, few people could've predicted the explosive growth of smartphones following the successful launch of the iPhone. Today, they are so ubiquitous that more people in India have access to smartphones than to toilets [55]. While these technologies have undoubtedly improved our lives in terms of convenience and connectivity, they have also brought with them a number of unexpected consequences.

For instance, some studies have demonstrated the so-called "Google effect", a consequence of having immediate access to a database of online information through your smartphone [56, 57]. Also called digital amnesia, it refers to the tendency to forget or misremember simple facts that can be easily found using search engines. The consensus on whether this change is positive or negative is mixed. At best, it may simply be the case that smartphone users are re-prioritizing what is and is not important to remember. At worst, it may be a sign that frequent smartphone users are losing or reducing their capacity to retain large amounts of facts and information, since they are so readily searchable.

Another example is the filter bubble phenomenon that has emerged throughout various corners of the internet [58]. It refers to a state of intellectual isolation caused by the personalization and selective presentation of content catered to what the user wants to see, without presenting opposing or disagreeing viewpoints. Events like the Facebook-Cambridge Analytica data scandal have brought to attention the potentially harmful effect of filter bubbles, suggesting they perpetuate media echo chambers and amplify the efficacy and spread of misinformation.

To date, filter bubbles are largely restricted to online platforms like search and social media. When we step outside into the real world, we cannot deny the existence of things that don't reflect our beliefs. But if we were to assume a future with perfectly lifelike augmented reality, then that may no longer be true. What I experience in one space may be entirely different from what you experience. I could, for instance, design an application that selectively subtracts things I don't want to see in the world, such as garbage, graffiti, or even homeless people on the street. State-of-the-art computer vision techniques can already generate realistic imagery to replace missing parts of an image, referred to as inpainting [59]. In this future, we only see what we want to see, which further perpetuates the information bubble and distorts our perception of reality.

Proponents of augmented reality predict a coming revolution in the personal computing space. If true, AR could potentially catalyze an even greater shift in how we process and consume information, by virtue of its constant presence, temporal immediacy, and ability to augment our perception of the world. It is imperative that we start understanding these effects earlier rather than later, lest we risk enabling and propagating poor behaviors and cognitive habits. As stewards of this technology, it is our ethical and moral responsibility to ensure it does more good than harm. AR systems should not de-emphasize the physical world in favor of the virtual world, lest users became overly reliant on AR systems to function in their everyday lives. Ideally, AR should improve human capabilities such that, even when the AR system is removed, the user has still gained tangible skills and benefits from their experiences.

The work of ensuring the ethical development of pervasive AR technologies starts with the people who create these systems. It is imperative that designers and engineers think critically about the systems we create and their impact on society from the beginning. AR practitioners need to think holistically about the technologies we choose to bring into the world and the benefits or drawbacks they may have. In my work, I address this challenge in two specific ways. I conduct experiments that analyze the positive effects of AR on the user, compared to existing modalities, providing a foundation for future designers to build upon. It is important that we understand fundamentally what benefits AR actually provides to the user through scientific validation, rather than let external forces like marketing or commercialization dictate them. A common application theme throughout this dissertation is foreign language learning, which I view as a potential “killer application” for pervasive AR. By taking advantage of the unique qualities of pervasive AR, such as the reinforcement of learning content via situated graphics, or continuous personalization for more efficient instructional scaffolding of learning content, we can demonstrate that pervasive AR systems can be both beneficial to its users and more effective than existing methods.

1.3.2 Gaps in Human and Computer Perception

There is a large gulf between the spatial representation capabilities of current state-of-the-art augmented reality techniques, and the spatial representations employed by users. AR vision systems locate physical surfaces on a three-dimensional Cartesian coordinate system. On the other hand, humans typically employ representational and semantic correspondences [60]. For instance, one might represent a room by its doors and windows, important markers for entry and egress that allow for quick decision making in case of emergency. This gap, between human-readable and machine-readable models of the same environment, make it difficult for developers to align their goals with the user’s aims [61] and is a significant barrier to development and iteration of AR applications.

Proponents of pervasive augmented reality often mention the necessity of integrating context-awareness and semantic understanding to AR systems, to the extent that some authors see context-aware AR and pervasive AR as the same concept [2]. Going from sporadic to continuous experiences invariably conflicts with our natural behaviors. We are fundamentally dynamic creatures whose conditions change and constantly evolve. Semantic understanding of situational and environmental context would seem a necessity to avoid information clutter and poor user experience. Understanding the user's context takes it one step further, allowing applications to align themselves to the user's behavioral and emotional state without being disruptive.

The definition of what exactly constitutes "context" has been in debate for hundreds of years. Even in the realm of computing and information systems, there are competing definitions depending on what level of abstraction you operate in. Computer Vision researchers for instance, might consider context to be any nearby visual phenomena that could be used to more accurately identify the target of interest. Systems researchers might consider context to include all nearby computing devices, and other technical resources such as power capacity, or the availability of certain wireless frequencies. In 1999, Dey and Abowd provided the most commonly cited definition used in HCI research [62]:

> "Context is any information that can be used to characterize the situation of an entity. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and applications themselves."

This definition is notable for its lack of specificity, instead defining context as a construct emerging from the actions and interactions of the user and the system. Through this lens, it can then be understood that user context is any information that characterizes the current state of the user during a particular interaction, which may or may not include the physical context or computing context, depending on the type of interaction. This definition also helps us understand other, less explored ideas of context, such as user goals and intentions, prior experiences with the system, or existing procedural knowledge. Similarly, we can define user context-awareness as when a system makes use of user context to present relevant information or services for a given task.

One of the principle challenges to understanding context is its temporality. The state of the user changes over time. User context doesn't exist in isolation, but is often informed by our past experiences. What characterizes a user's situation might, for instance, be influenced by their familiarity with the environment it takes place in. Within AR interactions, where computing engages the physical senses and occurs in the physical environment, perception and cognition also become relevant. After all, cognition is the way in which we organize and make meaning out of all the information, sensory or otherwise, that we accumulate over time. The ultimate challenge is to understand everything about the user up to their point of engagement with the system, customize the interface to meet their unique sensibilities, and provide an enriching AR experience symbiotic with the user and their dynamic circumstances. In a perfect world, that is what we would strive for. However, we don't live in a perfect world, but one where our personal data can be used for malicious intent. We need to understand how and how not to make use of user context, and ideally, allow the user to make choices around how much context to share for the tasks they want to accomplish.

Grubert et al. surveyed 96 papers featuring context-awareness in AR and categorized them into context targets (outputs) and context sources (inputs) [2]. Context sources were further classified into human factors, environmental factors, and system factors. Of these works, the majority of them focused on environmental context, likely a reflection of ocularcentrism in the research community. From works related to human factors, the authors only found one [63] that focused on the user's perception or cognition. Indeed, practitioners interested in user context will be left wanting.

Right now, most AR systems are static in their implementation, whether that be temporally static (i.e. designed to be used in one sitting), physically static (i.e. designed to be used in one location), or both. There is an immense amount of friction that prevents users from using

AR applications dynamically or continuously, as their operation conflicts with the reality of navigating their everyday lives. To build effective pervasive AR systems, we must understand users in all types of situations. That doesn't just mean environmental factors, but human factors as well.

As Dey and Abowd suggest, 'context' is heavily task-dependent and we should treat it as such. What are the user's intentions and goals? Where is their attention placed and how is it being used? How do users acquire, interpret, and represent situational knowledge? And how do they use it to make decisions about the environment? These are the types of capabilities we need AR systems to understand. In chapters 4 and 5 of this dissertation, I build upon these questions by exploring the feasibility of incorporating new task-dependent semantic understanding capabilities into AR.

1.3.3 Technology Monopolies and Data-driven Algorithms

Pervasive Augmented Reality will demand vast technological infrastructure. It is possible that such infrastructure, and ultimately the AR platforms and applications built on top of it, will be monolithic systems owned by a single or small group of stakeholders. At least, that is the vision being advocated by current proponents of "Metaverse" concepts such as Meta and Epic Games.

Tech monopolies do not always have our best interest at heart, but rather those of shareholders and the capitalistic drive for profit. This can be seen by numerous transgressions on data privacy and security, from Amazon giving out camera footage without user permission [64], to Facebook and the Cambridge Analytica data scandal [65]. It is not in our best interest as a society and as individuals to allow large corporations to have such an intimate relationship with our day-to-day lives.

In 2010, artist Keiichi Matsuda presented a provocative and arguably dystopian depiction of domestic augmented reality with his concept film Hyper-Reality. He portrays an augmented reality future in which our view of the world is oversaturated with virtual content, and every interaction is needlessly overcomplicated by a layer of technology. In Hyper-Reality, ever-present AR information is always available to assist you in any and every task, from the complex to the mundane, whether you like it or not. Content is inescapable as advertisers have your attention at all times, and they readily take advantage. The convenience of AR is now a crux for humans. Without it, you won't know which bus to take to get to work, or even worse, you might simply forgot to eat, drink, or sleep. Hyper-Reality represents the kind of excess we can expect when we drive technographic decisions using profit motives.

Notably, it is worrying that many of the biggest investors in AR are social media companies (Meta and Snapchat), whose profits overwhelmingly come from advertising (97.9% for Meta) [66]. That profit motive can incentivize companies to monopolize your attention in order to sell more advertising, with potentially negative consequences. For instance, the use of data-driven news feeds on social media platforms such as Twitter and Facebook led to significant increase in the propagation of fake news [58], with real world political consequences in many parts of the world. Their news feed algorithms prioritized user engagement metrics, rather than using human moderation or curation. As it turns out, fake news or antagonistic content that users disagreed with, actually kept them on websites longer, despite causing them unhappiness, loneliness, and displeasure [67].

Many proponents of pervasive AR have concluded that data-driven algorithms will ultimately be required to demystify and make sense of the massive amounts of data recorded by AR systems. This view is supported by significant improvements to the fields of Computer Vision and Natural Language Processing as a result of deep learning. It is unlikely that we will ever achieve comparable or better performance using traditional methods in the near future. To some extent, I agree with this perspective. The context awareness and computer perception challenges in the previous section will likely require data-driven solutions. However, in my opinion, data-driven algorithms should not be a replacement for user choice. The presentation of application content, the degree of augmentation, and the ultimate choice of which AR content to engage with, can and should be dictated by the end user. This doesn't mean that proactive interfaces don't have a place in AR, but rather, that there needs to be balance between the amount of data we give access to and the goals we want to accomplish with our AR applications.

This dissertation attempts to address this problem by exploring alternative AR application models and introducing novel methods for application switching and interaction. The app models we introduce place an emphasis on smaller-scale, service and functionality oriented applications, instead of the one-stop-shop model championed by large tech companies. These models have the additional benefit of distributing our personal and private data across multiple application developers, rather than storing it all in one place, improving our data privacy and reducing the effectiveness of data-driven algorithms.

In doing so, we set a precedent for research towards improving user autonomy, while developing an understanding of the value and usability of specific technological abstractions that can convey the dynamic capabilities of AR in a variety of settings. Technology does not develop in a vacuum, but is often inspired by things others have made before it. Hence, it is my hope that this work will encourage other designers, researchers, and technologists to consider privacy in future systems.

1.4 Permissions and Attributions

1. The work presented in Chapter 2 previously appeared in the proceedings of the 2018 IEEE International Conference on Artificial Intelligence and Virtual Reality [68], as well Volume 13, Issue No. 03 of the International Journal of Semantic Computing [69].

2. The work presented in Chapter 3 previously appeared in Volume 24, Issue No. 11 of the IEEE Transactions on Visualization and Computer Graphics [70].

3. The work presented in Chapter 4 were results of collaboration with Jason Orlosky at Osaka University. It previously appeared in the proceedings of the 2019 IEEE International Conference on Artificial Intelligence and Virtual Reality. It is reproduced here with his permission [71].

4. The work presented in Chapter 5 previously appeared in the proceedings of the 2019 IEEE International Conference on Virtual Reality and 3D User Interfaces [72, 73].

5. The work presented in Chapter 6 previously appeared in the adjunct proceedings of the 2020 IEEE International Symposium on Mixed and Augmented Reality [74].

6. The work presented in Chapter 7 is scheduled to appear in the proceedings of the 2022 IEEE International Symposium on Mixed and Augmented Reality.

Chapter 2

Effects on User Perception and Trust in a Recommender System

Pervasive AR presents significant human computer interaction challenges owing to its always-on nature. In theory, a pervasive AR system would be accessible and operating ubiquitously and continuously throughout our daily lives. The system would encounter a variety of everyday situations and have to adapt itself in such a way as to remain functional while remaining unobtrusive enough to allow the user to focus on the physical or real world task they need to accomplish. Current interaction paradigms may not be appropriate for achieving this level of seamless integration.

The interfaces we are familiar with today are typically reactive interfaces. They are reactive in the sense that they respond to user input through a mouse and keyboard or a touchscreen. They have no understanding of the user's goals or intentions, but only function through explicit user commands. In contrast to this are proactive interfaces. As the name implies, proactive interfaces anticipate what the user wants and may even execute them without explicit confirmation.

Reactive interfaces work well when computing is limited to a specific object or task. Computing is something you set out to do. You set aside time to do it and you engage with a specific object to accomplish it. As the walls between when we are “computing” and when we are “experiencing the world” shrink, reactive interfaces become problematic. By constantly having to convert our intentions into a series of interpretable computer commands, we create friction between our experience of the real world and the augmented world.

For this reason, some proponents of pervasive AR suggest the use of proactive, sometimes called implicit or noncommand user interfaces [75]. Instead of dictating what the computer should do, the computer would determine for itself what it thinks is appropriate. For instance, a wearable AR device might use camera and microphone sensors to interpret your context, determining that you are jogging, and immediately start tracking your health and fitness metrics and visualize the a recommended jogging path for your current fitness level. All of this happens without explicit interaction. The user simply decides to jog, and starts jogging.

Proactive interfaces are not common in the current computing landscape. AI and sensing technologies are not mature enough to feasibly design and evaluate large scale implementations of proactive AR systems. But one form of proactive interface that has been successful is the recommender system. Recommender systems proactively predict and show to users content the algorithm thinks the user will enjoy, and are widely deployed on e-commerce and social media platforms. Looking at recommender systems and how they are received in AR may tell us more about the effectiveness of future proactive AR interfaces.

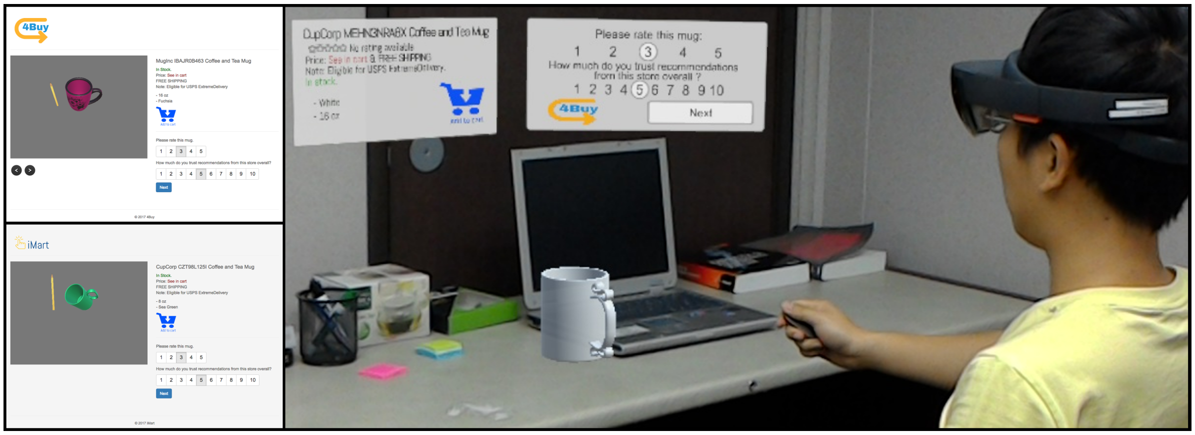
This chapter describes our efforts to create a product recommendation system in AR and evaluate users subjective perceptions and trust in the system. We implement and deploy the recommender system on a wearable AR headset and a web browser as control, and conduct a study comparing the two, intentionally varying aspects such as quality of the recommendations. The results provide novel insight into the ways participants perceive the AR content as part of the environment, and whether that effect is positive or negative in their reception of the interface.

2.1 Introduction

Recommender systems first emerged over two decades ago and have since become standard tools for dealing with information overload [76–78]. Major retail stores such as Amazon.com have a heavy focus on data-driven marketing, of which collaborative and content-based recommender systems are a core part. About 35% of sales on Amazon, and 75% of movies watched on Netflix are derived from recommendations [79]. The vast majority of recommendations for online retailers are delivered through email or in the traditional web browser interface. Interface technology, however, is developing rapidly: global revenues of Augmented Reality (AR) and Virtual Reality (VR) markets are expected to grow to over $162 billion in 2020 [80]. Heavy investment in AR and VR by major companies such as Apple, Alphabet, Facebook, and Microsoft will mean that smaller, higher quality devices will become available at lower cost to consumers. Large retailers such as Amazon and IKEA are exploring and introducing new AR driven shopping experiences.

While there has been progress on in-store AR technology to improve shopping experiences, e.g. [81], less work has been done on the concept of in-home shoppers taking advantage of what we call ‘situated recommendations’, whereby *personalized* recommendations of products are placed virtually where the real product will be used. In particular, we are interested in how people *perceive* recommendations that are situated in AR, and how this perception *differs* from that of traditional recommender system interfaces.

To answer these questions, we conducted a 3 by 2 within-subjects lab study (N=31). The study examined the effects of three different interaction modalities: an Augmented Reality interface, a web browser interface with 3D view controls, and a web browser interface with 2D view controls. We also looked at how users respond to differences in recommendation algorithm quality (either high or low quality recommendations). We measured two key metrics, user ratings of each recommended object (also called perceived accuracy), and user trust in the



4Buy

MugInc IBAJR0B463 Coffee and Tea Mug
In Stock.
Price: See in cart
FREE SHIPPING
Note: Eligible for USPS ExtremeDelivery
- 16 oz
- Fuchsia

Please rate this mug.
1 2 [✓ 3] 4 5
How much do you trust recommendations from this store overall?
1 2 3 4 [✓ 5] 6 7 8 9 10

© 2017 4Buy

iMart

CupCorp CZT98L125I Coffee and Tea Mug
In Stock.
Price: See in cart
FREE SHIPPING
Note: Eligible for USPS ExtremeDelivery
- 8 oz
- Sea Green

Please rate this mug.
1 2 [✓ 3] 4 5
How much do you trust recommendations from this store overall?
1 2 3 4 [✓ 5] 6 7 8 9 10

© 2017 iMart

CupCorp MEHN3NRA8X Coffee and Tea Mug
No rating available
Price: See in cart & FREE SHIPPING
Note: Eligible for USPS ExtremeDelivery.
In stock.
- White
- 16 oz

Please rate this mug:
1 2 [✓ 3] 4 5
How much do you trust recommendations from this store overall ?
1 2 3 4 [✓ 5] 6 7 8 9 10

Figure 2.1: Left: Screen captures of the browser interfaces. For 2D browser, users can look through photos of the mug taken from different predefined angles. For 3D browser, users can freely rotate the mug and view it in any direction. Right: Mixed reality screen capture of a user providing feedback to a recommended item.

recommender system. We collected subjective feedback on user perception of the modalities through a post study questionnaire and verbal interviews.

For the purposes of this study, we implemented a common online shopping user interface across all three modalities to allow for meaningful comparison. To avoid potential novelty effects, study participants undergo significant pre-study training sessions for each modality. Figure 2.1 shows an overview of our shopping interface. The right image shows a user wearing the HoloLens interacting with a virtual model of a recommended item and providing rating feedback to the system. The left images show the two web browser based interfaces that were tested in the study. In the web browser UIs, participants interact either by rotating the object with the mouse (3D), or clicking through static images (2D).

2.2 Related Work

Our study combines facets from multiple research fields, including human computer interaction (HCI), recommender systems, and cognitive science. A discussion of the relevant literature in each area is presented here, to frame our contribution in the context of existing research.

2.2.1 Augmented Reality in Retail Applications

Currently, there are many consumer applications for visualizing products in augmented reality. For example, IKEA uses a mobile AR app to place virtual models of their furniture in the physical world, with similar apps also coming from companies such as Walmart and Amazon [82]. Recent work by Stoyonova et al. [83] reports on a cognitive study of purchase intent using AR in a shopping scenario, but in contrast to this work, does not have a focus on personalized recommendations, and is situated in a store as opposed to a home shopping scenario. Lu et al. [84] perform a study of AR for home shoppers, where selected products can be tried in AR before purchase. Olsson and colleagues [85] present a study of user experiences with AR in a shopping center context and report mainly positive feedback for mobile AR supported shopping.

While there are many other examples of AR for improving shopping experiences [85, 86], to our knowledge there is no existing research that explores how users perceive *personalized recommendations* in this modality. We believe that our results can provide useful insight about this rapidly developing technology and its suitability as a channel for delivering personalized recommendations.

2.2.2 Augmented Reality and Recommendation

Many applications that integrate AR and recommendation use mobile platforms to perform location-based content recommendations. The Yelp monocle[1] for service recommendations is probably the most well-known example of this integration with AR. Balduini et al.'s Bottari system [87] provides personalized, location-based AR recommendations of social media

[1] https://www.yelp-support.com/article/What-is-Yelp-s-Monocle-feature?l=en˙US

content based on the Twitter network and evaluated the system in an urban area. While these approaches integrate AR and recommendation, they contrast with our approach in that they do not focus on evaluating perception of recommendations in AR compared to traditional UIs.

2.2.3 Interfaces and Decision Making

Prior research in recommender systems has a strong focus on algorithm performance. Recently however, more research attention is being paid to so called *user-aware* recommendation systems that attempt to improve the user's experience with the recommender system by mechanisms that go beyond predictive accuracy, such as conversation [88], explanations [77,89,90], and various different flavors of user interfaces [91–93], interaction designs [94] and evaluations [95–97].

In this study, we are interested in a novel user interfaces aspect –that of the impact of placement of recommended content in physical contexts with augmented reality, on the metrics of accuracy and trust. We are also interested in how the interplay of AI performance (quality of the recommendation) with the choice of user interface influences these metrics. It is likely that user specific factors such as experience with visualizations, recommender systems, or multimodal display technology will impact the observed results. Nilashi et al. [98] performed a mixed-model evaluation of recommender system users on two real world e-commerce sites and analyzed the impact of many observed and latent factors on trust and purchase intention. Similar mixed model evaluations for recommenders were performed on a hybrid music prediction system by Knijnenburg et al. in [97] and in a system for analysis of commuter traffic data from microblogs by Schaffer et al. [96]. In this paper we also apply a mixed-model evaluation, designed to capture user-specific characteristics that impact our performance metrics.

2.2.4 Trust Dynamics in Recommender Systems

Understanding and building user trust in predictions is an important goal of most recommender systems. Prior research has studied this from a computational model perspective to improve automated recommendations for collaborative filtering [99] and matrix factor approaches [100]. Others, such as [101, 102], have leveraged network information to build and propagate trust. In contrast to those relatively static approaches, we are interested in real-time human judgements of trust in both the system and its individual item predictions.

Recent work by Harman et al. [103] examines trust dynamics in a fictional and controlled online dating scenario under a repeated choice experiment with 200 trials. They found that users quickly learn to identify when poor recommendations are being made and lower their trust accordingly. An interesting aspect of their study looked at a personalized treatment against a non-personalized treatment and found that failures (poor recommendations) in the personalized condition had a more damaging impact on trust than in the non-personalized treatment. In our experiment design, we evaluated trust dynamics in a similar repeated choice and personalized scenario, but based on a simple home shopping task. A similar study by Yu et al. [104] also explores trust dynamics for an automated system under a variety of performance quality conditions. They find that increasing user familiarity with the system decreases the rate of change of trust after successes or failures of the automated system.

Building on the work from [103] and [104], we aim to explore the dynamics of trust for situated product recommendations in AR under conditions of high or low quality recommendations. Our study includes repeated interactions with the system to explore differences in trust dynamics and we hope to see trends similar to those found in the previous two approaches, with poor recommendation conditions showing less trust with each interaction.

2.3 System Architecture

To test our hypothesis, we implemented online shopping interfaces for each modality as well as a content recommendation system which generates a set of distinct high and low quality recommendations based on user profile data. These recommendations are distributed evenly across the three modalities, where they are rendered using the Unity game engine. During the study, users interact with each modality and give ratings which are sent back to the server to be recorded.

2.3.1 Browser Interface

We implemented a simple e-commerce graphical interface in a web browser (see Figure 2.1). The interface shows the recommended object, the store logo, and generic text descriptions of the object. The browser interface is broken up into two presentation modalities: 2D browser and 3D browser. In the 2D browser modality, item recommendations are presented as a set of 2D pictures of the product taken from different angles. Users cycle through these pictures by clicking on the arrow buttons below the image. A pencil is shown in the images to provide a point of reference for scale. Users can rate the items by clicking on the radio buttons provided on the right side of the image. The 3D browser modality displays a 3D model of item recommendations that users can interact with. In this modality, users click and drag within the display window to rotate the object about its central X and Y axes. Users can provide ratings in the same way as the 2D browser. Note that both kinds of browser interactions take place on a traditional computer and monitor.

2.3.2 Augmented Reality Interface

For the AR interface, we use a Microsoft HoloLens device. Our application uses the devices' Spatial Mapping API to map the environment and situate virtual products and UI el-

Table 2.1: Attributes for item classification.

Attribute	High-level choice	Value
Shape	Non-cylindrical	-1
	Cylindrical	1
Size	Small	-1
	Large	1
Color	Disliked	-1
	Neutral	0
	Liked	1

ements within the environment. We use HoloLens' World Anchor system to fix the recommended item and UI elements in the same position throughout the study. Users are able to walk around and look at the virtual items from different directions and provide feedback via the rating interface, presented through two panels as shown in Figure 2.1. The graphical interface is bare-bones, only displaying the store logo and generic text descriptions similar to the browser implementations.

For interacting with the interface, we implemented a 3D cursor using a raycast formed by the user's head gaze direction. We define head gaze direction as the forward direction of the headset. Using the 3D cursor, users can aim and click on the rating panel. Although the HoloLens device supports hand gestures for clicking, we opted to use the HoloLens bluetooth clicker. This provides a fairer comparison to the browser interface, as there may be additional effects introduced by gesture-based interaction.

2.3.3 Content Based Recommendation

In order to generate personalized recommendations, we use an algorithm based on attribute Preference Elicitation (PE) and Multi-Attribute Utility Theory (MAUT). The item attributes considered by the recommender are *color*, *shape*, and *size*. We provide a validation for this choice of attributes in the Experimental Design section. For each attribute, we compute the error between the recommender choice for that attribute, denoted as *recAttChoice*, and the user's preference, *userAttPref*.

If the attribute considered is a binary attribute (here, shape or size), let $recBinAtt \in \{-1, 1\}$ denote the recommender's choice for that attribute, where each possible value corresponds to a specific high level choice, as indicated in Table 2.1. The reported user preference for that attribute, *userAttPref*, has values in $[\![1, 5]\!]$. In the pre-study questionnaire, values of 1 and 5 corresponded to a strong preference toward one of the possible values of the binary attribute, values of 2 and 4 to a slight preference, and 3 to no preference. The error can be computed from those two variables as:

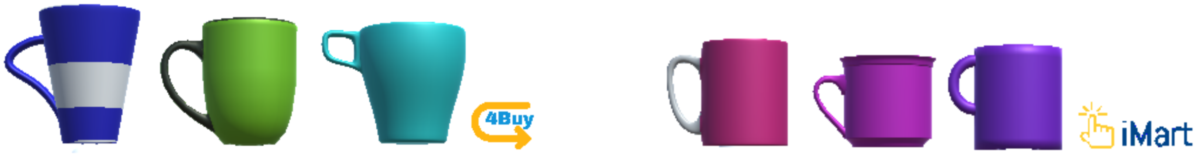
$$\mathrm{Err}_{recBinAtt} = L\left(recBinAtt, \operatorname{sgn}\left(userAttPref - 3\right)\right) \cdot W_{recBinAtt} \tag{2.1}$$

where $L(\hat{y}, y)$ is the 0-1 binary loss function which equals 1 if $\hat{y} \neq y$ and 0 otherwise, and the weight is defined based on the importance given by the user to that particular attribute as $W_{recBinAtt} = (userAttPref - 3)^2$. Note that 3 is subtracted from the user's rating in order to turn the values ranging from 1 to 5 from the pre-study questionnaire into values in $[\![-2, 2]\!]$. The sign function is then applied to map the user's rating to its corresponding value in Table 2.1. This essentially decouples the user's preference into a binary choice and a weight.

Color was treated slightly differently: users were asked how much color weighed in their decision, and then asked to choose colors they liked and colors they disliked among 13 colors; colors not selected are considered neutral. The estimation of the error on a given color choice by the recommender *recColChoice* therefore is:

$$\begin{multline}\mathrm{Err}_{recColChoice} = \left(\mathbb{1}_{DislikedCol}\left(recColChoice\right)\right. \\ \left. + 0.5 \cdot \mathbb{1}_{NeutralCol}\left(recColChoice\right)\right) \cdot W_{col} \tag{2.2}\end{multline}$$

where $\mathbb{1}_A(x)$ is the indicator function on set A defined as 1 if $x \in A$ and 0 if $x \notin A$. The weight



(a) Good recommendations

(b) Bad recommendations

Figure 2.2: Example of recommendations for a user indicating preference for large, non–cylindrical mugs with navy, lime, cyan as liked colors and indigo, magenta, fuchsia as disliked colors.

W_{col} has the same range of values as the weights used for the binary attributes, and the 0.5 factor for a neutral color ensures that picking a neutral color will yield an error superior to that of a liked color and inferior to that of a disliked color. The overall error is then obtain by summing the individual attribute errors:

$$\text{Error} = \text{Err}_{recColChoice} + \text{Err}_{recShapeChoice} + \text{Err}_{recSizeChoice} \tag{2.3}$$

It is worth noting that the errors can easily be turned into utility measurements by replacing L by $(1-L)$ in Equation 2.1 and $\mathbb{1}_{DislikedCol}$ by $\mathbb{1}_{LikedCol}$ in Equation 2.2.

The personalized recommender system computes all the possible values of the total error based on the user weights, and stores for each value of the total error the incorrect attributes contributing to that value of the total error. There are $2^{\text{card}(\{W_{BinAtt}:W_{BinAtt}\neq 0\})}$ possible ways to get an error from potentially incorrect binary attributes that the user indicated having a preference for, and $3^{L(W_{col},0)}$ different possible values for the error from the color. Note that the error is degenerate; that is, different choices of incorrect attributes may yield the same value of the total error, which can be used to diversify the recommendations. The recommender system can then use a look-up table to show products in order of increasing error.

We define *high quality* recommendations as ones for which every attribute satisfies the user's preferences, and *low quality* recommendations as ones maximising the error, i.e. none of the attributes satisfy the user's preferences. Extreme values of the error were picked to avoid issues with parameter tuning in the recommender algorithm (e.g. power used in the calculation of the weights), which may depend on the granularity of the preference scales or the different possible interpretations of the scale labels by the users. In a longer sequence of interactions, or real-world deployment of the system, less strict parameters could be adopted to improve diversity and novelty of predicted items. An example of the two classes of recommendations are shown in Figure 2.2.

2.4 Experiment Design

Our main study had a 3 by 2 within subjects design with counterbalancing. The two independent variables were *UI modality* and *algorithm quality*, and the main dependant variables were item ratings (accuracy) and user trust in the recommender system. A preference profile was gathered from each participant in the experiment several days before the in-person study, via a Qualtrics[2] online survey. In this preference elicitation questionnaire, participants were asked for basic demographic information and experience with recommenders and AR/VR technology. They were also asked to select preferences for each of the classification dimensions for our domain items. These consisted of size (large or small), mug shape (cylindrical or non-cylindrical) and color preference. For color, participants were shown images of 13 coffee mugs of different colors and were asked to select their favorite 3 and least favorite 3. This information was stored on a server which computed sets of high and low quality recommendations for each user, based on the algorithm previously described.

To compare the effect of recommendation quality among the three different modalities, two different virtual retail stores were created: *4Buy* and *iMart*. 4Buy always attempts to provide high quality recommendations and iMart always attempts to recommend items from the database that the user will dislike. Distinct logos for 4Buy and iMart were visible in each

[2] https://www.qualtrics.com/

Table 2.2: Validation of Item Classification Features from 110 participants in an online survey.

	Mean	Std Dev
Color	57.84	25.49
Pattern	58.45	27.5
Size	71.27	23.15
Shape	60.75	26.17

modality (see Figure 2.1) to allow users to recognize which store they are in and form different perceptions of trust for each store.

2.4.1 Item-space Classification

As it is difficult to find free high-quality 3D models, we chose to modify the models on the fly to provide variance in recommendations. We began with a total of 18 different models, and applied transforms over size and color parameters to provide different virtual mugs for participants. The patterns on the mugs varied. To ensure that the pattern variable would not impact user preference more than the controlled features (size, shape and color), an MTurk study of 110 users was performed where each participant provided ratings between 1 to 100 for each of the four features. The mean and standard deviation for these ratings are found in Table 2.2. We found no significant difference between pattern and the other features and so assume that manipulation of the other three features will be sufficient to provide good or bad recommendations based on the user profile. This is further confirmed in our results which show user ratings for good recommendations are significantly higher than those for bad ones. Different patterns in the mugs can contribute to *novelty* and *diversity* in recommendations, but overall quality can still be controlled in a meaningful way through manipulations on the other features.

2.4.2 Experimental Procedure

The experiment was conducted at an American university campus. Participants were assigned to particular orderings for each condition. Participants were given a brief introduction to the study by the experimenter. They were provided with a simple background story as follows: "You have just broken your coffee mug and are looking online to shop for a new one. You will shop at two different stores using a variety of their interfaces".

For the AR condition, participants were given a training task where they had to observe several virtual items and use interaction in AR to provide feedback ratings. Once comfortable with the AR environment and rating procedure, they began the main rating phase. Here, they were shown a sequence of three recommendations, either from iMart (low quality) or 4Buy (high quality). They were asked to walk around and inspect the items, and then provide a rating for how much they liked the recommended item on a scale of 1 to 5, and how much they trusted the system's current ability to provide good recommendations on a scale of 1 to 10. There was no time limit imposed during the rating phase. Participants typically took less than 30 seconds to provide a rating, irregardless of the modality. Similar training steps were performed for the browser-based conditions.

Participants complete all three conditions for a given store (and recommendation quality) first, before repeating the conditions in the same order for the other store. We alternated which store the participants start with. Once all conditions were complete, participants completed a post study questionnaire and were given a brief post study interview by the experimenter. In the post study questionnaire, participants were asked to rate their overall trust for each recommender, the helpfulness of the recommender, how much they liked interacting with each modality, and how much they liked each store overall. In the interview, participants were asked about their thoughts on the AR device, and whether they would choose to use it over the other modalities in a real world shopping scenario.

2.4.3 Novelty Effect

Since Augmented Reality is a new and emerging technology, and there is a "wow factor" with cutting edge devices such as the Hololens, novelty effects will always be challenging to deal with. To mitigate novelty effects in the experiment, participants were allowed up to 10 minutes to familiarize themselves with each modality. In the AR condition, participants played with the built-in holograms application on the Hololens device. Note that this familiarization period takes place before the training task begins.

After the experiments, we compared performance between the participants who started with the AR condition versus those who started with the browser-based conditions. We ran paired t-tests on our key metrics but found no significant differences between the two groups, giving us confidence that our balancing and familiarization procedures were helpful in controlling novelty effects of the Hololens device in the AR condition. This was further supported through post study interviews, during which participants reported that the familiarization period helped them to "get comfortable" using the AR headset.

2.5 Results

To answer our research questions, we looked at user ratings for individual product recommendations and overall trust in the recommender system. We examined differences in ratings across each modality in order to assess relevant effects on user's perception of recommendations. Additionally, we examine self-reported UX metrics from a post study questionnaire and verbal interview.

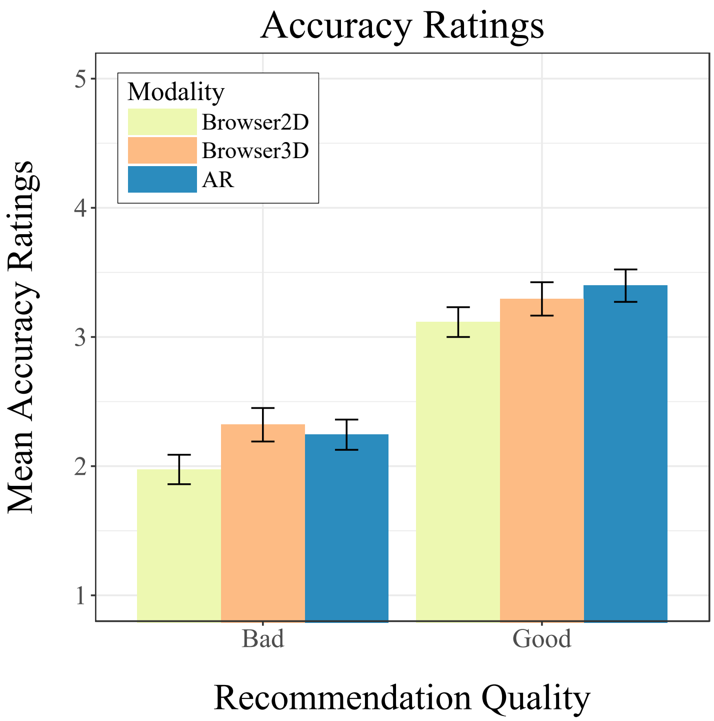


Figure 2.3: Mean accuracy rating with standard error.

2.5.1 Participants

In total, 31 participants completed the in-person study. Data from three participants were removed due to being provided incorrect instructions on the rating system. These participants misunderstood the task and rated other aspects such as the design of the logo. We also removed two additional participants due to system failure of the HoloLens during the experiment, leaving a total of 26 for analysis. Participants had a median age of 23, mean age of 27 with std. deviation of 9.58. 77% were male and 23% female. All had at least some college education. Participants were recruited through a user study pool at the university and were paid $10 for the study, which lasted about 40 minutes.

Table 2.3: Accuracy: Pairwise comparison between modalities.

contrast	estimate	SE	df	t.ratio	p.value
Browser2D - Browser3D	-0.2641	0.1095	440.51	-2.413	**0.0428**
Browser2D - AR	-0.2791	0.1091	438.92	-2.557	**0.0293**
Browser3D - AR	-0.0150	0.1091	438.42	-0.138	0.9896

Results are averaged over the levels of: Recommendation Quality

P value adjustment: Tukey method for comparing a family of 3 estimates

2.5.2 Perception of Product Recommendations

Our first research question focuses on the perception of the products recommended by the system. To begin our analysis we looked at the average accuracy ratings within each condition. The resulting data is graphed in Figure 2.3. We tested for significance using paired t-tests.

For these ratings, our initial hypothesis was that increased reality and immersion provided by the AR modality would amplify users' perception of recommendation accuracy. More realistic inspection methods might cause users to have a greater awareness of how well a product fits their preferences. Thus, we expected bad recommendations to be rated lower in AR compared to browser based methods, and likewise good recommendations would be rated higher in AR.

When looking at ratings in the bad recommender, we found a significant difference between the 2D modality ($\mu = 1.97$) and the 3D modality ($\mu = 2.32$) conditions; p = 0.024. There was almost significance between 2D modality and the AR modality ($\mu = 2.24$); p = 0.064. In the good recommender, we found significance between the 2D modality ($\mu = 3.12$) and the AR modality ($\mu = 3.4$); p = 0.035, but not between 2D and 3D modalities.

Additionally, we see a significant difference in ratings between the good and bad recommenders for all three modalities (all $p < 0.0005$). This gives us confidence that our recommendation algorithm is correctly providing high and low quality recommendations based on the

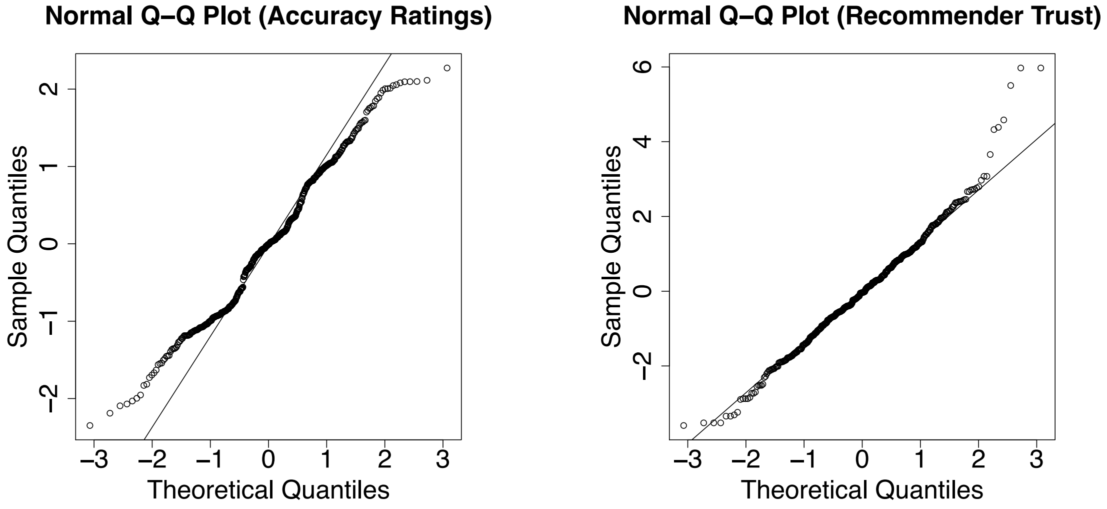


Figure 2.4: Q-Q plots of residuals of LME models for accuracy and trust.

user's preferences.

These results appear to reject our hypothesis. Irregardless of recommender quality, AR and 3D modalities seem to improve perception of recommendations. However, the signal does not appear consistently between bad and good recommenders. Thus, to look at the effects of each modality across both good and bad recommender conditions, we opted to perform further analysis using linear mixed effects models. Specifically, the modality type and recommendation quality are modeled as fixed effects, while participants and item design are modeled as random effects.

To validate this approach, we assessed the fit of our models using pseudo-R^2 values [105]. Marginal pseudo-R^2 was computed for fixed effects, and conditional pseudo-R^2 for random effects. For the accuracy model, the marginal pseudo-R^2 was 0.216 and the conditional pseudo-R^2 was 0.369. Additionally, mixed effects models assume that the residuals of the model are normally distributed. We plotted the residuals of each model as Q-Q plots to check this assumption and found that the residuals fall about a fairly straight line, suggesting normality. These plots can be found in Figure 2.4. Finally, we created separate models where Modality and Recommendation Quality were modeled as having an interaction effect. We performed a likelihood ratio test against these to determine any significant interaction effects, but did not find any significant inter-dependence between them thus we did not include interaction effects in our models.

The full pairwise comparisons between each modality are shown in Table 2.3. These tables describe the difference in ratings after averaging over the levels of recommendation quality and performing p-value adjustment using the Tukey method. Here, we can see a significant difference between Browser2D and the AR modalities (p = 0.0293), as well as between Browser2D and Browser3D (p = 0.0428). This provides further evidence that AR may improve user perception of product recommendations.

When comparing AR against the 3D interface, pairwise comparisons within our model did not show a significant difference in product rating. We believe that this result was due to a hidden variable created through differing levels of control in the interaction. In the 3D browser, users could rotate the items and view them from all angles. However, in the AR condition, the item was in a fixed position, and therefore could not be viewed from the bottom angle, since it was positioned on a table. During the verbal interview, three participants mentioned they prefer the 3D view because it “allows you to see the mug in every possible orientation”.

2.5.3 Perception of the Recommender System

The next research question focuses on the perception of algorithm quality within the different modalities. Similar to the product ratings, we hypothesized that the AR condition could help improve user awareness of a recommendation algorithm’s performance, leading to lower ratings for the low quality recommender and higher ratings for the high quality recommender compared to the other modalities.

Our analysis on trust ratings mirrored the methods used for product ratings in the previous

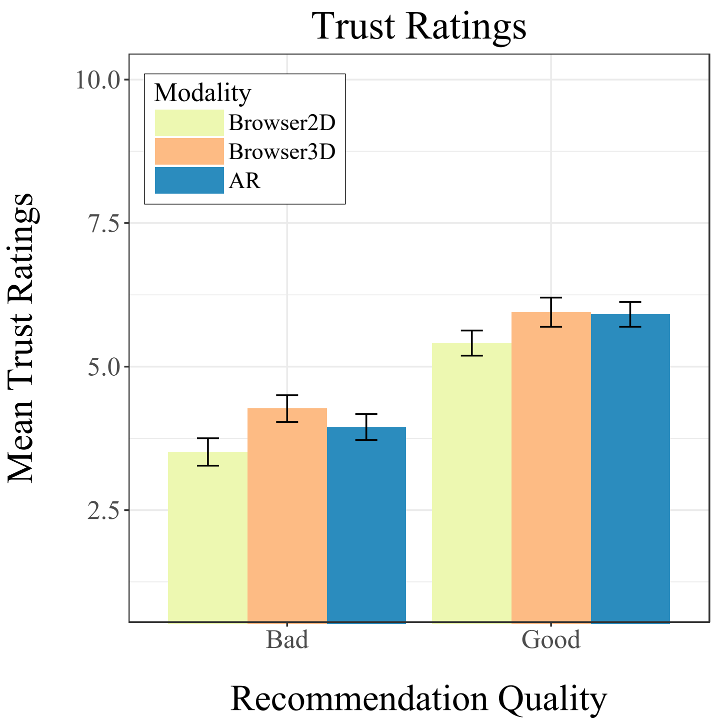


Figure 2.5: Mean trust ratings with standard error.

Table 2.4: Trust: Pairwise comparison between modalities.

contrast	estimate	SE	df	t.ratio	p.value
Browser2D - Browser3D	-0.6474	0.1697	442	-3.815	**0.0005**
Browser2D - AR	-0.4679	0.1697	442	-2.757	**0.0167**
Browser3D - AR	0.1795	0.1697	442	1.058	0.5410

Results are averaged over the levels of: Recommendation Quality

P value adjustment: Tukey method for comparing a family of 3 estimates section. In Figure 2.5 you can see the graphed trust ratings. In the bad recommender, we found significant differences between 2D ($\mu = 3.51$) and 3D ($\mu = 4.27$); $p < 0.001$, as well as 2D and AR ($\mu = 3.95$); p = 0.034. In the good recommender, we also found significance between 2D ($\mu = 5.41$) and 3D ($\mu = 5.95$); p = 0.005, and also between 2D and AR ($\mu = 5.91$); p = 0.008. Again, we see a significant difference between the good and bad recommenders for all three modalities (all $p < 0.0005$). Figure 2.5 clearly show that users perceived a difference between good and bad algorithms in all conditions. For example, participants in the 2D browser condition rated trust in the iMart (low quality recommender algorithm) at 3.51 and 4Buy at 5.41, which is a relative improvement of 54% over the iMart algorithm.

We again used linear mixed models to analyze trust ratings across recommendation quality. We performed the same steps to validate the model as in the previous section. For the trust model, marginal pseudo-R^2 was 0.184 and conditional pseudo-R^2 was 0.555. Table 2.4 is the resulting pairwise comparisons. In particular, we highlight the differences between Browser2D and the AR modalities which are significant for trust ratings (p = 0.0167). Ultimately, the results we found did not support our hypothesis. Instead, our results suggest that Trust is improved for the AR and 3D modalities, despite the differences in recommender quality.

Trust Dynamics

We build on recent work in recommender systems research by examining the perception of trust in the recommender system over time, for the high and low quality recommendation algorithms. We plot these trends for each modality in Figure 2.6. The first clear effect from this is the separation between the high and low quality recommendation strategies (4Buy and iMart). This provides further support of the effectiveness of our recommender system, despite its relative simplicity.

Looking at the slopes of these distributions, all but one of the data points for the low quality recommender (iMart) follow a downward sloping trend, while those for the high quality recom-

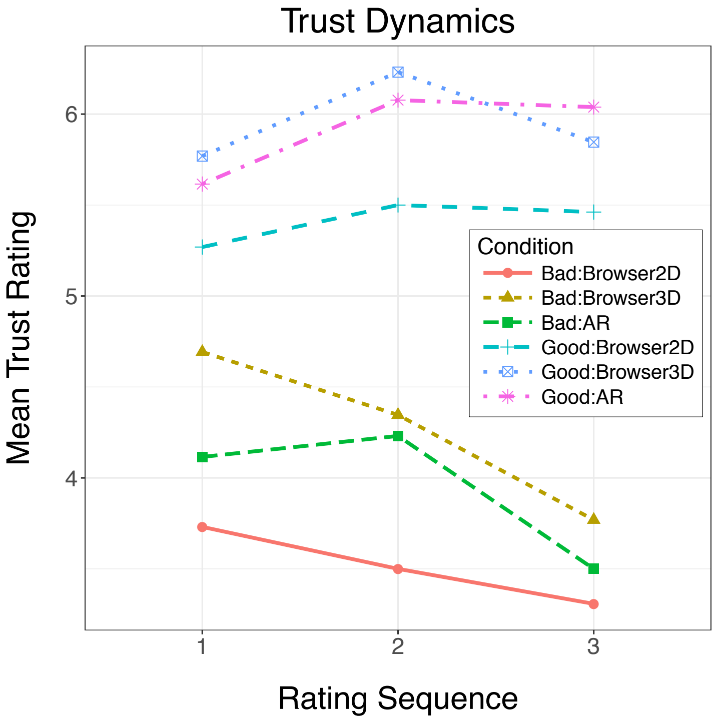


Figure 2.6: Dynamics of trust for each condition.

mender (4Buy) have an initial upwards trend. This supports similar results found in [103], in which users trust in the system dropped swiftly following repeated interactions with poor recommendations. This is further supported by our post study questionnaire, where participants significant preferred 4Buy over iMart.

Additionally, we see a decrease in the rate of change of trust after repeated interaction, between the first and second recommendation to the second and third recommendation. However, this is only present for the good recommender system. This trend is similar to results found in [104].

For further analysis, we used Analysis of Covariance (ANCOVA) to compare trust ratings categorized by condition, controlling for rating sequence. Our test did not find any significant interaction between rating sequence and condition (p = 0.515), suggesting that there aren't any significant differences in the slopes of the regression lines between each condition. We

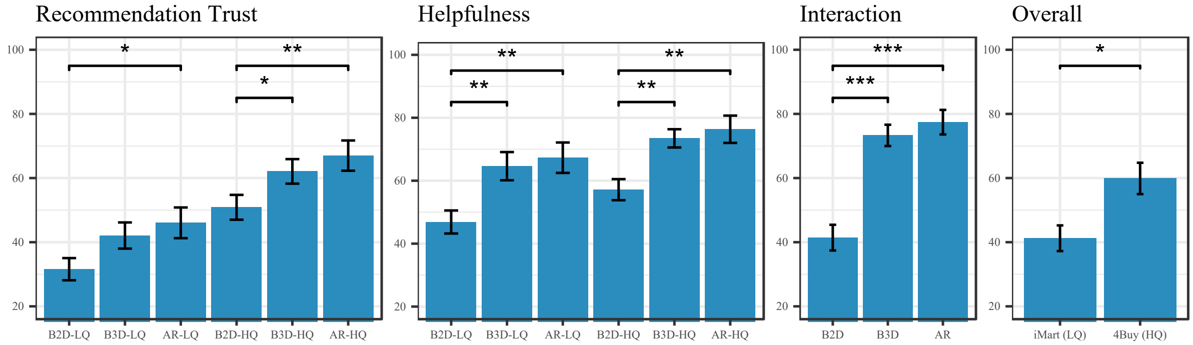


Figure 2.7: Mean subjective ratings from the post study questionnaire with standard error. Participants were asked to rate how much they trust the recommendations, how helpful each store's interface was, the interaction quality of each modality, and overall preference for each store. Brackets show the level of significance between particular values (* p<0.05, ** p<0.01, *** p<0.001). Additionally, there was significance (p<0.01) in recommendation trust between each HQ modality and their LQ counterparts.

believe this may be due to the limited amount of repeated interactions. Additionally, we looked at whether trust changed over time for high and low quality recommendations regardless of modality. We analyzed the average ratings for the first and last modality used for both the low and high quality recommender using a paired t-test, but did not find a significant difference in either case (p = 0.783).

2.5.4 Participant Sentiment

Our last research question focused on the general sentiment of participants towards an AR recommender system for in-home shopping. Our primary source of analysis for this research question are through a post-study questionnaire and semi-structured verbal interview conducted immediately after the experiment.

Post-Study Questionnaire

The results of the post questionnaire are shown in Figure 2.7. The leftmost plot shows the perceived trust in the system's recommendations broken down for each of the six conditions.

Here the browser-based conditions are abbreviated to B2D and B3D, and algorithm quality is represented as HQ or LQ for high and low respectively.

The first point to note is that the questionnaire responses for trust in the recommendations align well with the observed ratings during the experiment, with both AR-HQ and B3D-HQ showing a significant rating improvement of about 20% over the traditional UI B2D-HQ. There was no significant difference between the B3D and AR conditions. However, our post study interviews revealed that people either had a strong preference for the 3D-browser condition or the AR condition. Those who strongly preferred AR, tended to mention the value of being able to see the item in real-world context (situated recommendations), while those who preferred the 3D browser version tended to like the familiarity of the interface for shopping.

Perceived helpfulness of the stores was also evaluated and showed a similar trend to trust, with AR and B3D having significant rating improvement over B2D for both recommender algorithms (LQ and HQ). However, the differences between recommender quality (LQ and HQ), was not as pronounced as it was on the trust metric. We believe this is an indication that users were considering other aspects than recommendation quality for their decisions on helpfulness, such as the quality of the UI design.

Figure 2.7 also shows results for perceived quality of interaction with the system. As expected, both AR and B3D received very positive ratings. This is consistent with our interview feedback where participants preferred B3D almost as much as the AR condition due to better inspection capabilities. Participants were also asked to rate each store overall. Here we see that participants did perceive the difference in algorithm quality across the two stores. The store with high quality recommendations (HQ) showed a 50% improvement over the LQ store. This was consistent with our observed ratings-based results.

Verbal Interview

Participants were given a verbal interview immediately after the post-study questionnaire, which typically lasted about five minutes. For the verbal interview, we provided some structure by asking in order the following questions:

1. What did you think of the HoloLens?
2. Would you use this in a real world shopping scenario?
3. Did you find the ability to walk around and view objects in a real world environment to be helpful or distracting in your shopping decisions?

Participants were asked all three questions regardless of if they had already mentioned any related comments in a prior question. This means that some participants touch on the same topic multiple times over the course of the interview.

When asked what they thought of the HoloLens, participant opinions were generally quite high. Most participants thought the AR device was cool, interesting, and enjoyable. Even before being prompted in question 3, participants often commented about the ability to walk up to the object, see it from different angles, and compare the object to its surroundings. The most common negative opinion was the color fidelity of the display. Five participants had complaints about colors being washed out and difficult to perceive. Other complaints include the limited field of view, and discomfort due to weight of the device.

When asked about whether they would consider using the AR interface in a real world shopping scenario, participants responses were very positive. All but five of the participants reported that they would choose to use the AR system if it were available to them. Out of many different reasons cited, the most common was the ‘try before you buy’ reason –to visualize and interact with the item in the context where it is to be used. An equally common opinion was the desire to use the interface for purchasing certain types of items. Typically, participants

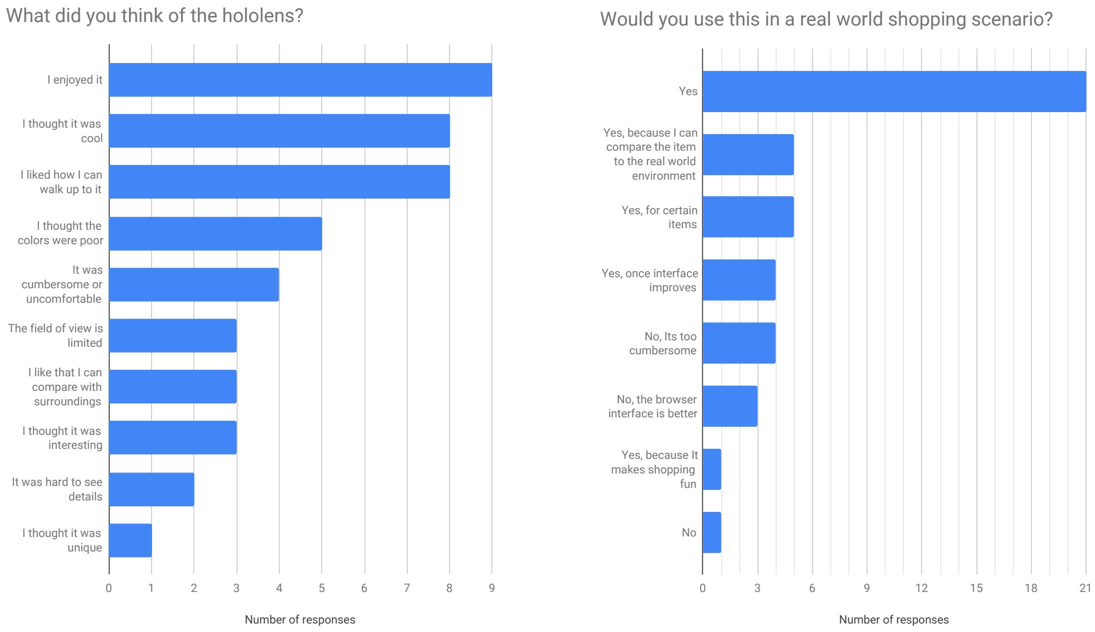


Figure 2.8: Summary of common responses in the verbal interviews for questions 1 (left) and 2 (right).

mentioned it would be very useful for purchasing large items such as furniture. A few participants commented that they would use AR shopping once the interface was improved. In this case, they felt the interface was very useful for shopping but wanted a more "polished" user interface design. The participants who did not want to use it argued that the interaction was not sufficient and that the 3D browser version allowed for a better inspection of the item. Additionally, participants reported feelings of frustration and discomfort that would dissuade them from using the device.

A summary of the most common responses to the first two questions can be found in Figure 2.8. Note that participants may have commented on multiple topics during the course of answering each question.

For the third question, we were able to bucket the responses into four categories. 13 participants said the ability to view products in-situ was very helpful, while seven participants said it was only slightly helpful. Five participants said it was neither helpful nor distracting to the shopping task. Only one person said it was somewhat distracting to see while shopping. This same participant ultimately commented that they preferred the browser version because it was much faster and more efficient to use.

Ultimately, these responses show a lot of positive sentiment for the future of AR recommender systems. Our study participants don't view in-situ shopping as more distracting or more difficult. The biggest concerns came from hardware limitations or lack of polished designs, issues that would surely be addressed by future commercialization efforts and design improvements that were not the focus of our controlled test interfaces. Already, recently deployed or announced products such as the MagicLeap One[3] or HoloLens 2[4] are lighter, more comfortable, and have twice the field of view. These additional technological capabilities should lead to better audience acceptance regarding these issues.

2.5.5 Demographic Analysis

We wanted to look at potential differences between demographics in their experience of the AR recommender system. We recorded participant demographics and looked at mean trust and accuracy ratings between each demographic group. We grouped participants by age, gender, and their familiarity or prior experience with AR devices.

When looking at gender, we had 20 males and 6 females take part in our study. Our analysis showed that females gave higher product ratings ($\mu = 2.96$) in AR conditions when compared to males ($\mu = 2.66$); p = 0.045, using Welch's t-test. However, there were no significant differences for mean trust ratings. Additionally, females reported having less familiarity with the modality than males. When asked on a scale of 1 to 5 about their prior experience with AR, the mean for females was 3.12 compared to 3.4 for males.

To look at age, we grouped participants into two age bins around the median, and a similar

[3] https://www.magicleap.com/magic-leap-one

[4] https://www.microsoft.com/en-us/hololens

analysis was performed. We performed similar t-tests for mean ratings and mean trust but did not find any significant differences between the two age groups. Additionally, older participants tended to have more prior AR experience, with an average rating of 3.21 vs. 3.0 for the younger participants.

Finally, we separated participants based on their familiarity with AR into two groups: those with low or no experience with AR, and those with some prior AR experience. The low experience group was composed of the 12 participants whose prior experience with AR was 1 or 2 out of 5, leaving 14 participants for the other group. Using Welch's t-test, we found that participants with little AR experience had significantly higher product ratings ($\mu = 2.91$) compared to those with some AR experience ($\mu = 2.57$); $p < 0.005$. They also had higher trust ratings ($\mu = 5.21$) compared to those with more AR experience ($\mu = 4.51$); $p < 0.005$.

To help explain these results, we look to participant comments in their post study questionnaire. Participants who have more AR experience tend to be more critical of the AR device's limitations, noting things like the weight of the device or the poor resolution of the display. Whereas those who are newer to the interface are more excited about its potential, and are more willing to forgive these faults.

2.6 Discussion

This chapter presented a study that to our knowledge is the first empirical analysis of the effects of Augmented Reality interfaces on the perception of recommender systems. A 3 by 2 within subjects experiment assessed user perception of high and low quality personalized recommendations in three modalities: Augmented Reality with recommended items placed in a real world scene, web browser with 2D images, and web browser with 3D interaction. Quantitative metrics for product ratings and recommender trust were assessed, along with perception of the system through a post study questionnaire and verbal interview.

Results of our main research questions show that overall product ratings for recommended objects, and trust in the recommender, are significantly higher in AR and interactive 3D than in a traditional browser UI. However, there is no significant difference in either metric between interactive 3D and AR modalities. Furthermore, people perceive differences between high and low quality algorithms in all three modalities, but there is no significant trend that suggests better awareness of quality differences in AR. Finally, a majority of participants preferred to use AR over browser based interfaces for product recommendations, finding it helpful for visualizing in the context where it will be used.

The results from our study contribute to an emerging body of work focused on understanding user perception of AR with proactive AI systems such as recommender systems. Throughout the study, AR and Browser3D modalities performed on par with each other, whereas both tended to improve ratings and other metrics compared to Browser2D. Participants generally fell into two camps, those preferring Browser3D and those preferring AR.

Many of the verbal interview responses seem to indicate that participants appreciate qualities from both mediums. In the case of AR, participants enjoy being able to visualize a product in a real world context and grasp the actual scale of the object. However, AR is marred by issues with a low quality display and headset discomfort. 3D on the other hand is quick and easy to use, and still allows users to view recommended products from a variety of viewing angles.

While some of these problems will be solved in future iterations of AR devices, it is important to understand what the role of interaction should be moving forward. It is clear that some users actively preferred the browser based interaction methods. For many shopping experiences, and possibly other types of tasks, it is likely that users will prefer methods they are accustomed to over an AR experience. Where AR has the potential to excel is when offering an experience that has tangible real world implications, such as when delivering recommendations that have great impact on daily life, or where scale and contextual information can be compared in a real life setting, such as home appliances and interior design. These qualities should be emphasized and communicated when designing for the future of AR driven recommender systems.

Chapter 3

Effects on Learning and Memory

In the previous chapter we saw that AR was not always the most preferred interface. Depending on the task, many users might prefer to a different style of interface because it might allow them to focus better or provide a more efficient set of controls for accomplishing the task. For AR to grow and attract users organically, we need to ask ourselves, what benefits are unique to pervasive AR.

Thinking about the problem from an augmented human perspective, pervasive AR can be seen as an extension of our visual processing capabilities, since it is consistently and continuously rendering an altered view of the world for us to process. If that is the case, then it is possible that augmented content could incorporate itself into the cognitive processing loop of visual perception. This brings up serious implications for pervasive AR. As detailed in the first chapter of this dissertation, we need to be careful not to deskill users by acting as a permanent crutch or replacement for existing skills, and we need to be careful not to enable users with rose colored lenses that show the world as they prefer to see it, rather than as it truly is. But that doesn't stop us from taking advantage of the positive aspects of close integration with our daily lives and sensory processing.

For example, if we persistently see and associate new information in AR with features of the physical environment, does it then lead to improvements in learning and memory? There is some evidence to suggest that, with the right training, we can improve our memory and retain significantly more knowledge by incorporating spatial cues [106, 107] in our learning process. A highly effective technique familiar to every memory champion and polyglot is the memory palace, in which the desired information is arranged alongside the layout of some building, street, or other geographical entity. Can we use AR to provide everyday users with some of the advantages of a trained memory palace user, even without any training?

In this chapter, we explore the potential of AR to facilitate improved memory by associating information with physical objects arranged in an unfamiliar environment, comparing the situated AR language learning interface against a tablet-based flash card interface mode. In this work, we collaborated with educators from Gervitz Graduate School of Education to carefully craft a curriculum for learning vocabulary words in the Basque language, and conduct a within subjects study to discern benefits to recall, both immediately after the study and delayed, several days after. In addition, we lay out the foundation and long-term vision for AR language learning as an ideal pervasive AR use case. We carry this theme forward and continue to iterate on it in later chapters.

3.1 Introduction

This paper addresses the problem of facilitating and understanding the process of language learning in immersive, augmented reality (AR) environments. Recent heavy investment in AR technology by industry leaders such as Google, Microsoft, Facebook and Apple is an indication that both device technology and content for this modality will improve rapidly over the coming years. Looking forward, we believe that AR can have significant impact on the way we learn foreign or technical languages, processes and workflows, for example, by creating new personalized learning opportunities in a physical space that is modeled, processed and labeled

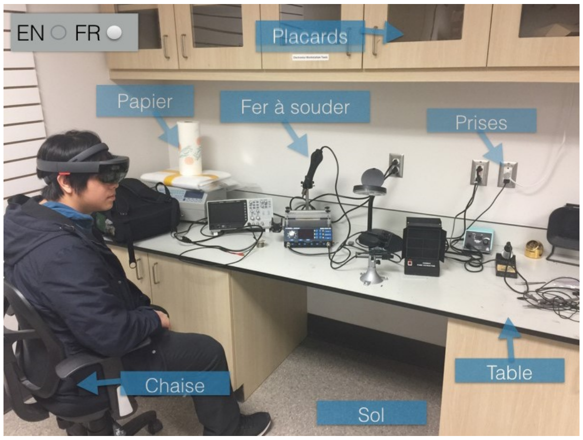


Figure 3.1: An illustrative mock up of a language learner using the ARbis Pictus system. Note that the user will only see annotations in an approximately 30° × 17° field of view, due to current constraints with the HoloLens.

by automated machine learning (ML) classifiers, assisted by human users. These augmented learning environments can include annotations on real objects, placement of virtual objects, or interactions between either type to describe complex processes. Thus, without significant extra user effort, a future "always-on" AR system could seamlessly provide a user with the foreign-language terms describing objects (or later possibly even processes) in their own physical environments, enabling casual reminders and incidental learning of vocabulary.

Such learned language skills will still be in effect and valuable when the AR device is not in use. This is in contrast to using automatic translation services, which can also strongly benefit from AR technology but which require being online and may not actually help in *learning* a language if relied upon without reflection, much like unconsidered reliance on online navigation services might not improve navigation skills, and in fact can lead to inferior spatial knowledge [108].

AR devices will eventually become affordable and portable enough to be commonly used in day-to-day tasks. In this setting, learning can occur passively as people interact with objects and processes in their environments that are annotated to support personalized learning objectives.

Our project towards this vision is called ‘ARbis Pictus”, named after Orbis Sensualium Pictus (Visible World in Pictures), one of the first widely used children’s picture books, written by Comenius and first published in 1658. To study the benefits of using AR for memorization tasks such as used in language learning, we conducted a foundational user study to evaluate the impact of learning simple noun terms in a foreign language with augmented reality labeling using a purpose-built AR object labeling tool. While the scenario given above for the medium-to-longer-term vision includes the potential for casual incidental learning, this first study examines active intentional learning efforts. We focus on the following three research questions in our user study:

- RQ 1: When learning vocabulary (or individual lexical items) in an unknown second language, is there a benefit of learning with AR over a traditional flashcard-based method?
- RQ 2: In the above setting, how does productive recognition and recall vary after some time has passed?
- RQ 3: How do users *perceive* the language learning experience in Augmented Reality compared to traditional flashcards?

In the process of answering these research questions, we make the following contributions, results and insights. To carry out our study, we designed and implemented a system that supports foreign language vocabulary learning with augmented reality and with traditional flashcards. We then designed and implemented a user experiment to evaluate the impact of AR-based learning for foreign language vocabulary. Key findings include 1) better recall (7%, $p<0.05$) for AR learning compared to traditional flashcards; 2) an increased advantage (21%, $p<0.01$) for AR in productive recall four days after the initial test, compared to traditional flashcards; 3) Qualitative survey and interview data shows that participants believe that AR is effective and enjoyable for language learning. During the study, attention and gaze data was collected through the AR device, through an eye tracker, and through click interactions. We also describe an early-stage analysis of this data, and how it reveals learning patterns in each modality.

3.2 Related Work

Our study aims to show that there is a measurable benefit to learning vocabulary through AR labeling of real world objects. Before we discuss the design, we briefly describe existing work on multimedia learning and on AR in education settings.

3.2.1 Multimedia Learning

Our framework is motivated by Mayer et al.'s cognitive theory of multimedia learning (CTML) [109] [110], one of the most compelling learning theories in the field of Educational Technology. The theory posits, first, that there are two separate channels (auditory and visual) for processing information, second, that learners have limited cognitive resources, and third, that learning entails active cognitive processes of filtering, selecting, organizing, and integrating information. The CTML predicts, based on extensive empirical evidence, that people learn better from a combination of words and pictures than from words alone [111]. In the field of Second Language Acquisition, studies using the CTML as their theoretical basis have shown that when unknown vocabulary words are annotated with both text (translations) and pictures (still images or videos), they are learned and retained better in post tests than words annotated with text alone [112] [113] [114]. A second principle of the CTML is that people learn better when corresponding words and pictures are presented near rather than far from each other on the page or screen, as the easy integration of verbal and visual information causes less cognitive load on working memory, thereby facilitating learning [115]. Second Language Acquisition research has found that simultaneous display of multimedia information leads to better performance on vocabulary tests than an interactive display [116]. Our approach extends the CTML by simultaneously displaying information next to physical objects, allowing learners to further integrate spatial information of the object and its surroundings.

A recent study by Culbertson et al. in [117] describes an online 3D game to teach people Japanese. Their approach used situated learning theory, and they found excellent feedback on engagement. Specifically, people were learning eight words in 40 min on average. Experts who already knew some Japanese were the most engaged with the system. Learning results from that study informed the design and complexity of the learning tasks in our experiment. The broader vision for our ARbis Pictus system, including personalized learning and real-time object recognition was influenced by work by Cai et al. in [118], which found that we can leverage the small waiting times in everyday life to teach people a foreign language, e.g. while chatting with a friend electronically.

3.2.2 Virtual and Augmented Reality in Education

The use of Augmented Reality for second language learning is in its infancy [119] [120], and there are only a small number of studies that link AR and second language learning. For example, in [121], Liu et al. describe an augmented reality game that allows learners to collaborate in English language learning tasks. They find that the AR approach increases engagement in the learning process. In contrast, our experiment is an evaluation of the effects of immersive AR on lexical learning, using simple noun terms only, analogous more to a traditional flashcard-based learning method. The benefits and shortcomings of flashcards are well documented in the second language learning literature [122]. In this study, we employ this method as a simple benchmark, purposely chosen to minimize effects of user interactions, and to expose the impact of immersion in AR. Grasset et al. [123] and Scrivner et al. [119] have studied AR textbooks in the classroom. Their approach differs from ours in that we use minimal virtual objects (labels only), but incorporate physical objects in the real world as a pedagogical aid, including their spatial positioning in the augmented scene. Godwin-Jones [120] provides a review of AR in language learning, focusing on popular games such as Pokemon Go! and general AR devices and techniques, but doesn't discuss formal evaluations of AR for second language learning. Going beyond simple learning of lexical terms, the European Digital Kitchen project [124] incorporates process-based learning with AR to support language learning. They apply a marker-based tracking solution to place item labels in the environment to help users prepare recipes, including actions such as stirring, chopping or dicing, for example. Dunleavy et al. [125] discuss AR and situated learning theory. They claim that immersion helps in the learning process, but also warn about the dangers of increased cognitive overload that comes with AR use. In our experimental design, we consider this advice and allow ample time for familiarization with the AR device to reduce both cognitive overload resulting from the unfamiliar modality, and other novelty effects.

AR has been shown in first experiments to support better memorization of items [126], [127], making use of spatial organization and the memory palace technique. Our study is in line with these promising results and shows that there can be a distinct benefit of AR for vocabulary learning, comparing with tried-and-true flashcard-based approaches.

Another benefit of AR is that it brings an element of gamification to the learning task, making it particularly suitable for children to learn with. There have been several interactive games involving AR for learning in a variety of situations. Costabile et al. [128] discuss an AR application for teaching history and archaeology. Like [125], they hypothesized that engagement would be increased with AR compared to more traditional displays. However, the results found that a traditional paper method was both faster and more accurate than AR for the learning task. Another notable example is Yannier et al.'s study [129] on the pedagogy of basic physical concepts such as balance, using blocks. In their study, AR outperformed benchmarks by about a 5-fold increase, and was reported as far more enjoyable. A similar, but much

Order	*Device used — word subgroup(s) seen during each learning phase*					
I	AR - A1	AR - A1, A2	AR - A1, A2, A3	FC - B1	FC - B1, B2	FC - B1, B2, B3
II	AR - B1	AR - B1, B2	AR - B1, B2, B3	FC - A1	FC - A1, A2	FC - A1, A2, A3
III	FC - A1	FC - A1, A2	FC - A1, A2, A3	AR - B1	AR - B1, B2	AR - B1, B2, B3
IV	FC - B1	FC - B1, B2	FC - B1, B2, B3	AR - A1	AR - A1, A2	AR - A1, A2, A3

Table 3.1: Table of conditions and balancing across the six learning phases. AR shows the augmented reality conditions and FC represents flashcards. A and B are distinct term groups for the within-subject design, and the group number indicates one of the subgroups of 5 words.

earlier approach that applied AR to collaboration and learning was Kaufman's work [130] on teaching geometry to high-school level kids. An updated version of this system was applied to mobile AR devices by Schmalstieg et al. in [131]. Now that we have described relevant related work that has informed our experimental design and setup, we can proceed with details of our designs. This will be followed with a discussion of results.

3.3 Experimental design

52 participants (33 females, 19 males, mean age of 21, SD of 3.8) took part in a within-subject study. Learning modality and word groups were systematically varied, resulting in a 2x2 counterbalanced design. Participants were recruited through a paid pool at a university, and were a mix of students and non-students.

In terms of ethnic background, 16 participants self-identified as White or Caucasian, 14 as Asian, 9 as Hispanic or Latina/Latino, 3 as Asian and Caucasian, 4 as Latino/Latina and Caucasian, 3 as African American, 1 as African American and Mexican, 1 as Middle Eastern, and 1 as Mixed without further elaboration. No participant had high proficiency in three or more languages. 19 participants reported being natively or fully professionally proficient in a second language. Of the other 33 participants, four reported speaking more than one language with intermediate proficiency, and 24 with some elementary proficiency. 16 people reported having some kind of proficiency in three or more languages.

English was by far the most commonly well-spoken language, with 48 participants reporting “Native or Multi-lingual Proficiency” in it, and four participants “Full Professional Proficiency”. The second most commonly well-spoken language was Spanish, with 30 participants reporting having any kind of fluency in it. The third most spoken language group consisted of Chinese languages, as reported by 9 participants. No participant reported any kind of proficiency in Basque, the language of choice for our experiment.

30 Basque words were divided into two groups of 15, called A and B, further divided into fixed subgroups of 5 referred to as A1, A2, A3 and B1, B2 and B3. Each subject saw one of the two word groups on one of the devices, and the other group on the other device. In total, 13 people saw the word group A in AR first, 13 word group B in AR first, 13 word group A with the flashcards first, and 13 word group B with the flashcards first, as described in Table 3.1.

After answering a set of background questions, participants were told what the objects were in English and started using one of the devices after the instructors informed them about the learning tasks and the specifics of the tests. On the AR device, the participants first undertook a training task where they could take as much time as they wanted to set up the device, get used to the controls and reduce the novelty aspect of it while interacting with virtual objects. Before using the flashcards, an eye-tracker was calibrated for each participant. Then, the participants moved on to the learning task, which consisted of 3 learning phases and 4 tests (3 productive recognition tests, 1 productive recall test) per device. In the first learning phase, the participants had 90 seconds to learn the 5 words of the first subgroup of one of the word groups on a given device. After a distraction task, they took a productive recognition test. Afterwards, they undertook a second learning phase on the same device, and had 90 seconds to learn the 5 new words from the second subgroup of the same word group, along with the 5 previous words. Following a distraction task, they took a recognition test on the 5 new words. They then had a third learning phase on the same device during which they saw for 90 seconds the 5 words of the last subgroup of the selected word group, alongside the 10 previous ones. After a distraction and a recognition test on the 5 words from the last subgroup, they took a productive recall test on all 15 words from the word group chosen. They then had another, similar set of 3 learning phases and 4 tests on the other device using the other word group, as illustrated in Table 3.1. The AR learning task, flashcard learning task and tests took place in three different rooms to avoid potential biases.

At the conclusion of the learning task, the participants answered a questionnaire on how efficient and engaging they perceived each device. A short interview allowed us to gather more feedback on their preferences. Four days after the learning phases, the participants were asked to take again the same eight tests they took the day of the study to assess long-term recall. 32 users agreed to take the tests.

Every participant was compensated $10, and the study lasted a total of 40 to 65 minutes for every user (with most of the variance due to the AR training phase's flexible length).

3.4 Experimental setup

Our experiment setup consisted of two modalities. An AR learning tool and a traditional flashcard tool, implemented as a browser-based web application. We now present the details of both, along with the learning and distraction tasks that were completed by each participant.

3.4.1 Flashcards

The flashcard modality was designed as a web application emulating traditional physical flashcards, running on a desktop computer that the user interacted with using a mouse (see Figure 3.2). After entering a user ID and one of the combinations of word subgroups seen in Table 3.1, the instructor let the participants interact with 1, 2 or 3 rows of 5 flashcards, all visible on a single page, with each flashcard consisting of a word in the foreign language on the back and an image of the corresponding object on the front. The images used were pictures of the real

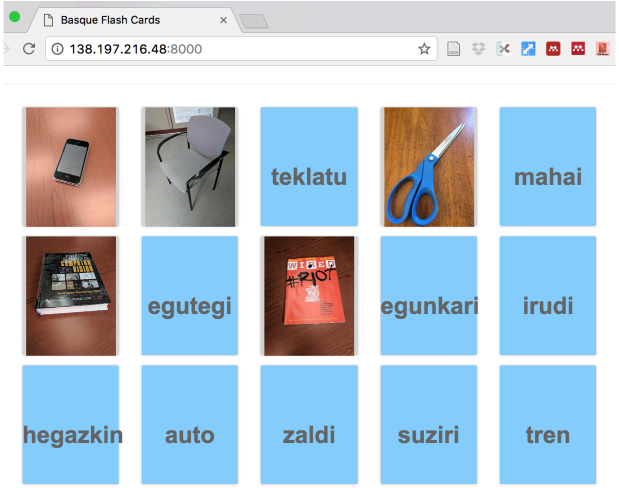


Figure 3.2: Screen shot of the web-based flashcard application that was used in the study.

objects used in the AR condition. A recording of the word being pronounced was automatically played through speakers every time the user clicked on the back of a flashcard. The same recording of the Basque word being spoken by a human (male) was used in both modalities. Clicks were logged during every phase to track possible learning strategies. Additionally, an eye tracker was calibrated before the learning task with the flashcards to track the participants' gaze during the learning phases.

3.4.2 Augmented Reality

The augmented reality modality (shown in Figure 3.3) made use of a Microsoft HoloLens, an augmented reality head-mounted display. The application was set up in a room containing all of the objects from the two word groups, but only allowed the participants to see labels annotating the objects from the currently chosen subgroups with the relevant words in the foreign language. The device's real-time mapping of the room let the users walk around the room while keeping the labels in place, and save the location of the labels throughout the study, between users and after restarting the device. As a precaution, before every learning phase on

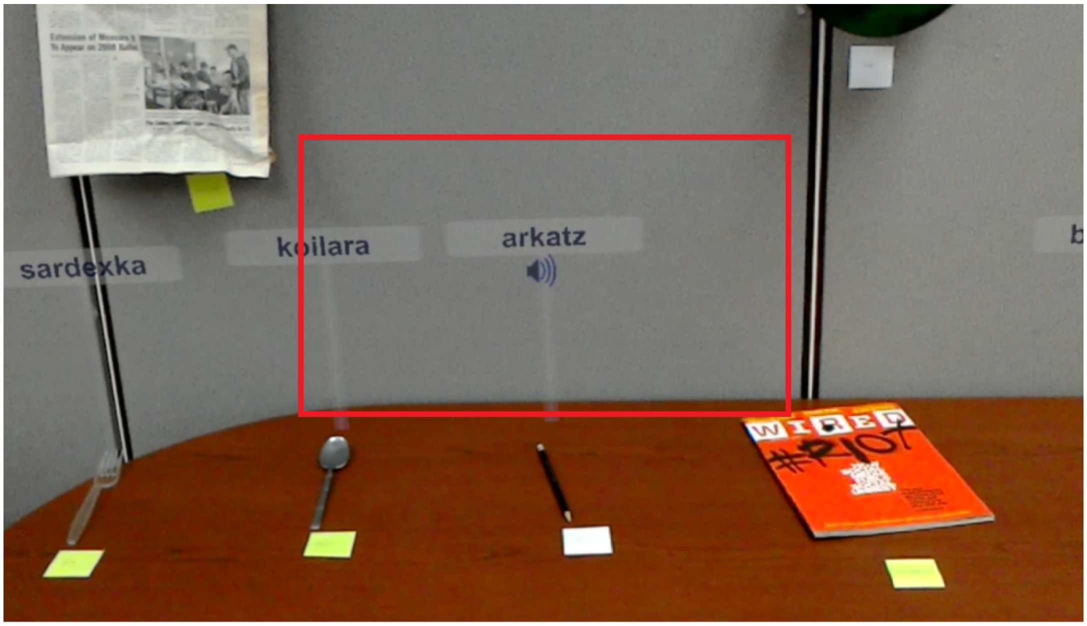


Figure 3.3: Example of the Basque labels shown in the AR condition of the experiment. This is a HoloLens *mixed reality capture* of the scene, which exaggerates the AR field of view. The approximate actual field of view for the label annotations is highlighted in red.

the HoloLens, the administrators of the study verified that the labels were in place, and after handing over the device to the participants, that they were able to see every label. The app had two modes: “admin mode”, allowing the instructor to place labels with voice commands or gestures, select which word subgroups to display, or enter a user ID; and a “user” mode that restricted these functionalities but allowed the participants to interact with labels during the learning task.

On the HoloLens, the cursor’s position is natively determined by the user’s head orientation; in the app, moving the blue circle used as a cursor close to a label would turn the cursor into a speaker, signalling to the user the possibility to click to hear a recording of the word being pronounced through the device’s embedded speakers. Each label had an extended, invisible hitbox to allow the users to click the labels more comfortably. Moreover, the labels’ and hitboxes’ sizes, along with the real objects’ locations, were adjusted based on the room’s dimensions and the device’s field of view to ensure that the participants could not see more than two labels at the same time, and that looking at a label would most likely lead to the cursor being in that label's hitbox. This was used to log the attention given to each word during the learning task, in "user mode".

In between the learning phases, "admin mode" was switched on to display a new subgroup, check on the labels, and temporarily disable logging of attention data.

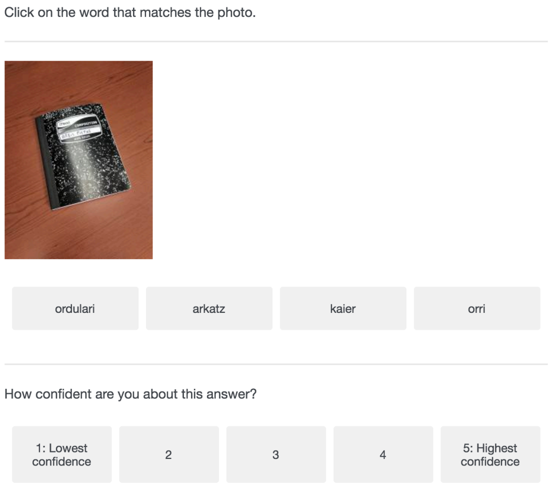
Due to the HoloLens's novelty, the participants were allowed to interact with animated holograms for as long as they wished before the AR learning task to get used to the controls, adjust the device and overcome some of the novelty factor of the modality.

3.4.3 Learning Task

The Basque language was chosen after ruling out numerous languages that shared too many cognates (words that can be recognised due to sharing roots with other languages) with English, Spanish and other languages that are commonly spoken in the region where the study was administered. Basque presented interesting properties: Latin alphabet to facilitate the learning, but generally regarded as a language isolate from the other commonly spoken languages [132], allowing us to control the number of cognates more easily, with one of the authors being fluent in the language. The 30 words were carefully chosen and split into two groups A and B based on difficulty and length, and further split into three subgroups per word group where each subgroup corresponded to a topic: A1 was composed of office related words (pen, pencil, paper, clock, notebook), A2 of kitchen related words (fork, spoon, cup, coffee, water), A3 of clothing related words (hat, socks, shirt, belt, glove), B1 of some other office related words (table, chair, scissors, cellphone, keyboard), B2 of printed items (newspaper, book, magazine, picture, calendar), and B3 of means of locomotion (car, airplane, train, rocket, horse). The study's counterbalancing helped address possible issues arising from A and B potentially not being balanced enough. The learning task on a device was constituted of three learning phases, each of which lasted 90 seconds, for a total of two learning tasks (one per device) or six learning phases across both devices. The limit of 90 seconds was adjusted down from 180 seconds after a pilot study had shown a large ceiling effect with the users reporting having too much time. Once A or B was chosen as a group of words, the users successively saw subgroup 1 (5 words) during the first learning phase, then subgroups 1 and 2 (10 words) during the second learning phase, and then 1, 2 and 3 (15 words) in the last learning phase. The decision to allow the users to review the previous subgroups came as a solution to avoid the floor effects in the productive recall tests observed in the pilot study.

3.4.4 Distraction Task

In order to prevent the users from going straight from learning to testing, a distraction was used to reduce the risk of measuring only very short-term recall. The task needed to have enough cognitive load to distract the participants from the words they had just learnt. The participants' performance at the task should also be correlated to their general performance regarding the study, in order to avoid introducing new effects – for example, a mathematical computation may bias the results as a participant with above average computational skills but below average memory skills may pass it fast enough that they would perform as well as another participant with below average computational skills but above average memory skills. Therefore, the distraction was chosen to be a memorisation task, in which the participants were asked to learn a different alphanumeric string of length 8 before every recognition test. The six codes used were the same for everyone, and were presented in the same order for every participant for the 2x2 balancing to mitigate possible ordering effects.



Click on the word that matches the photo.

How confident are you about this answer?

Figure 3.4: Format of the Productive Recognition Test.

3.5 Metrics

To best understand the lasting impact of learning in the different modalities, our metrics included production and recognition of vocabulary, both immediately after a learning session, and also in test several days afterwards.

3.5.1 Productive Recognition Test

The productive recognition tests were administered on the desktop computer used for the questionnaire, in a different room from the two used for the learning tasks. Figure 3.4 shows the format of the test. The questions consisted in 5 images, each accompanied by a choice of 4 words from which the participants had to pick the appropriate one. Each image corresponded to one of the 5 new words seen in the preceding learning phase: A1 or B1 after the first learning phase, A2 or B2 after the second learning phase, and A3 or B3 after the third learning phase, depending on which one of A or B was chosen as the word group for that learning task, for a total of 6 recognition tests across the two learning tasks. All 5 images were available on the same page, allowing the participants to proceed by elimination. There was no time constraint, to avoid frustrating the participants, who were encouraged to use the tests as a way to prepare for the productive tests due to the strong floor effects observed in the pilot study. The performance was measured as either 1 for a correct answer, or 0 for an incorrect answer. Every question was accompanied by a confidence prompt on a scale of 5 ranging from “Lowest Confidence” to “Highest Confidence”.

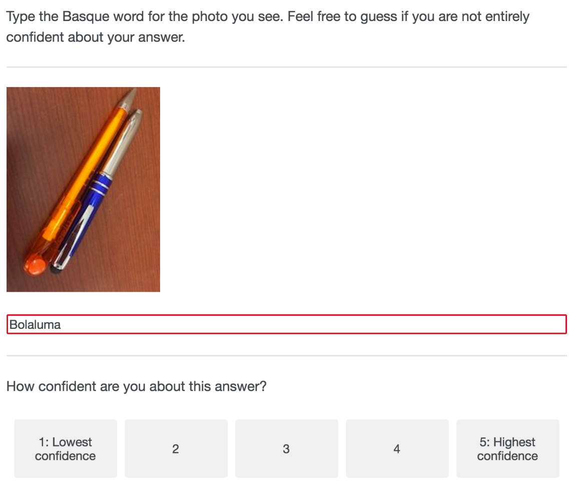
3.5.2 Productive Recall Test

The productive recall tests took place on the same computer used for the recognition tests, immediately after the third recognition test at the end of each learning task. Figure 3.5 shows the format of the test, which also required a confidence evaluation for each answer. The productive recall test had 15 images corresponding to the 15 words from the selected word group, and participants were asked to type the corresponding word in Basque below each image. The error on a participant’s answer was measured using the Levenshtein distance, which counts the minimum number of insertions, deletions and substitutions needed to transform a word into another, between their answers and the correct spellings [133]. Participants were therefore encouraged to try their best guess to get partial credit if they did not know the answer, and had to provide an answer to every question to end the test. The Levenshtein distance was also upper bounded in our analysis by the length of the (correctly spelled) word considered, to prevent answers such as “I don’t remember” from biasing a participant’s average error, and divided by the length of the correct answer to get a normalised error:

$$\mathrm{AdjLev}(w,\hat{w}) = \frac{\min(\mathrm{Lev}(w,\hat{w}),\mathrm{Length}(\hat{w}))}{\mathrm{Length}(\hat{w})} \tag{3.1}$$

where w is the participant’s answer on a given question, and $\hat{w}$ the correct answer. The score was then computed as

$$\mathrm{Score}(w,\hat{w}) = 1 - \mathrm{AdjLev}(w,\hat{w}) \tag{3.2}$$



Type the Basque word for the photo you see. Feel free to guess if you are not entirely confident about your answer.

Bolaluma

How confident are you about this answer?

Figure 3.5: Format of the Productive Recall Test.

where 1 indicates a perfect spelling, and 0 a maximally incorrect answer. As in the recognition tests, every question was accompanied by a confidence prompt on a scale of 5 ranging from "Lowest Confidence" to "Highest Confidence".

3.5.3 Delayed Test

The delayed tests consisted of the same tests used for the same-day testing, in a slightly different order: the productive recall test of each word group was administered before the three recognition tests to prevent participants from reviewing with the recognition tests due to the absence of a time constraint. The tests were sent in a personalized email to the participants four days after the study. Only tests completed in the 24 hours after receiving the email were kept in the analysis. Further, the test did not allow the participants to press the back button, and only tests completed in a similar amount of time as the same-day tests were kept. Participants were informed that the study being comparative, the absolute number of words they remembered did not matter, and that the goal of the study was to measure how many people performed better with either device with no expectation of a modality being better than another. This was done in

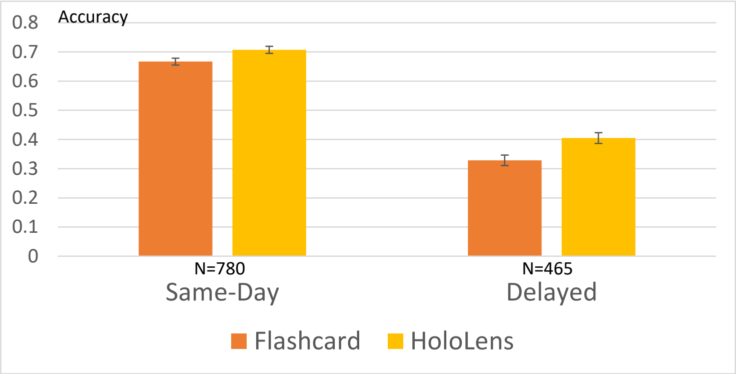


Figure 3.6: User performance on same-day and 4-day-delayed productive recall tests. The left group shows the flashcard and AR accuracy score for the same-day test and the right side shows the comparison for the 4-day-delayed test. Error bars show standard error here.

order to reduce the impact of potential demand effects such as bias towards either modality, and only their remembrance of the words (i.e. no qualitative feedback) was evaluated in the delayed test to further diminish such biases. In total, 31 participants' delayed test answers satisfied the criteria mentioned above. Note that the 2x2 counterbalancing was conserved (eight participants had followed order I, II and IV and seven order III as defined in Table 3.1).

3.6 Results

Now that we have described our experimental setup and metrics, we present a discussion of results, organized by test type (recall and recognition), a brief discussion of attention metrics (gaze and click behavior), and a discussion of qualitative feedback from post-study questions and interviews.

3.6.1 Productive Recall

Figure 3.6 shows the accuracy results of the same-day productive recall test compared to the delayed test for both modalities. The AR condition is shown in the lighter color. The

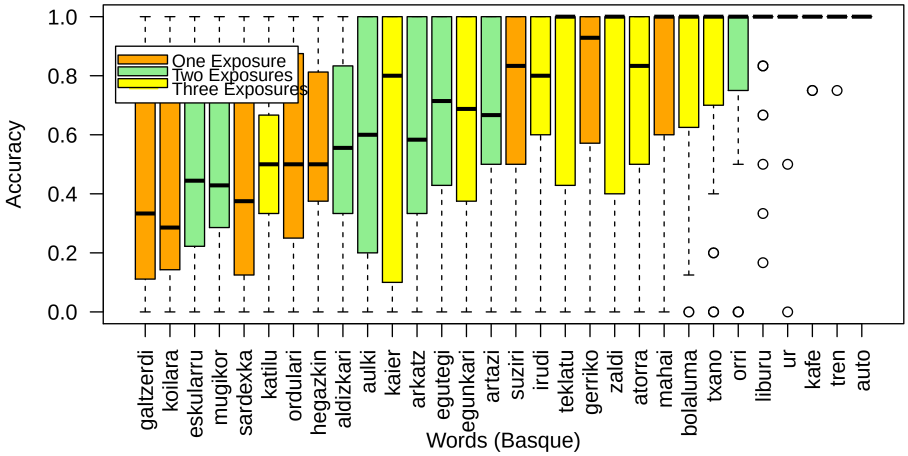


Figure 3.7: User performance on delayed productive recall tests, ranked by term. Colors show exposure groups. The accuracy score on the y-axis is computed from the mean of the normalized Levenshtein distance between the participants' spelling and the correct spelling.

delayed test was administered four days after the main study, and there was some attrition, with 780 question responses in the main study and 465 for the delayed. Accuracy was measured using the score function previously defined in Eq.3.2 as 1 minus the normalized Levenshtein distance between the attempted spelling and the correct spelling. In the same-day test, the AR condition outperformed the flashcards condition by 7%, and more interestingly, in the delayed test, this improvement was more pronounced, at 21% better than the flashcard condition. The test results were analyzed in a non-parametric way after Shapiro-Wilk tests confirmed the non-normality of the data. This is due in part to the many occurrences of words perfectly spelled. Both differences are significant with Wilcoxon Signed-rank tests: p=0.011 and p=0.001 for

Dependent Variable (Accuracy)	Z	*p*	effect size
Same-day Prod Recall	-2.5397	**0.01109**	0.352
Delayed Prod Recall	-3.1959	**0.001394**	0.574
Same-day Recognition	-0.7926	0.42799	0.110
Delayed Recognition	-0.1239	0.901389	0.022
Same-day Prod Recall (FC pref group)	1.1589	0.246488	0.237
Delayed Prod Recall (FC pref group)	-0.0580	0.95367	0.016

Table 3.2: Key results from statistical analysis. Results highlighted in bold face are statistically significant effects.

the same-day and the delayed productive results respectively, as seen in Table 2. The table also reports productive recall scores for those users who reported that Flashcards were more effective than AR (FC pref Group). Interestingly, no significant difference was found between the modalities for this sub-group, in contrast to the results for the general population.

Based on interviews with the participants, we believe that the significant improvement in delayed recall is linked to the spatial aspect in the HoloLens condition. Several participants reported qualitative feedback to this effect, such as in the following example: *"One reason the AR headset helped me recognize the words better is because of the position of the object. Sometimes, I'm not memorizing the word, I'm just recognizing the position of the object and which word it correlates to. "*

3.6.2 Productive Recognition

Productive recognition was analyzed in the same manner as productive recall, however a histogram of response accuracy revealed a ceiling effect in the data, where many participants provided fully correct responses. The mean productive recognition score was 0.89 for the same-day test in both modalities and 0.84 in the 4-day delayed test again for both modalities. In the delayed test, the productive recall was presented first to avoid learning effects from viewing multiple choice options. There was no significant difference between the modalities in this test.

3.6.3 Attention Metrics

Gaze data was gathered for both modalities. For the flash card application, we collected eye tracking data using a screen-based eye tracker, and for the HoloLens application we recorded head orientation focus as described above.

For each of the terms in the three different exposure groups we computed the average time that participants' attention was focused on that item. This was performed primarily to examine why repeated exposure to terms did not produce an observed improvement in accuracy. For the first group, the mean was 13.5 seconds (SD 6.3 seconds), for the second, the mean was 10.8 seconds (SD 7.2 seconds), and third group had a mean attention time of 7.2 seconds (SD 4.5 seconds). The differences in attention times not being significant between the different groups may imply that during the learning phases, participants focused mainly on the new items, or that users chose to focus on different words on average. This is reinforced by the fact that no significant accuracy improvement was measured for repeated-exposure items, as evidenced by Figure 3.7 where the most significant effect is word length.

Click data was recorded for the flashcard application to help identify potential learning patterns. Recall that the flashcards had two sides and required a click to turn from text to image and back again (Figure 3.2). The click patterns showed that people tended to click more often towards the end of the study. 18 of the participants had a pattern of clicking the same flashcard over five times in a row, perhaps indicating a desire to see both image and text at the same time, or testing themselves during the learning phase. Both possibilities are supported by users reporting in the post-study interview that they enjoyed the ability to see the object and the word simultaneously in AR, while others mentioned making use of the flashcards' two-sided nature to self-test.

3.6.4 Perception and Qualitative Feedback

Participants were asked about their experience using AR and flashcards, and their subjective ratings correspond with their learning performance. In terms of what was fastest for learning words, 54% found AR fastest, compared to 46% who found flashcards fastest. As a side note, 13 among the delayed test population had reported preferring the flashcards, as opposed to 18 for AR. As for the learning experience, 75% of participants rated AR "good" or "excellent", while 63% rated flashcards "good" or "excellent".

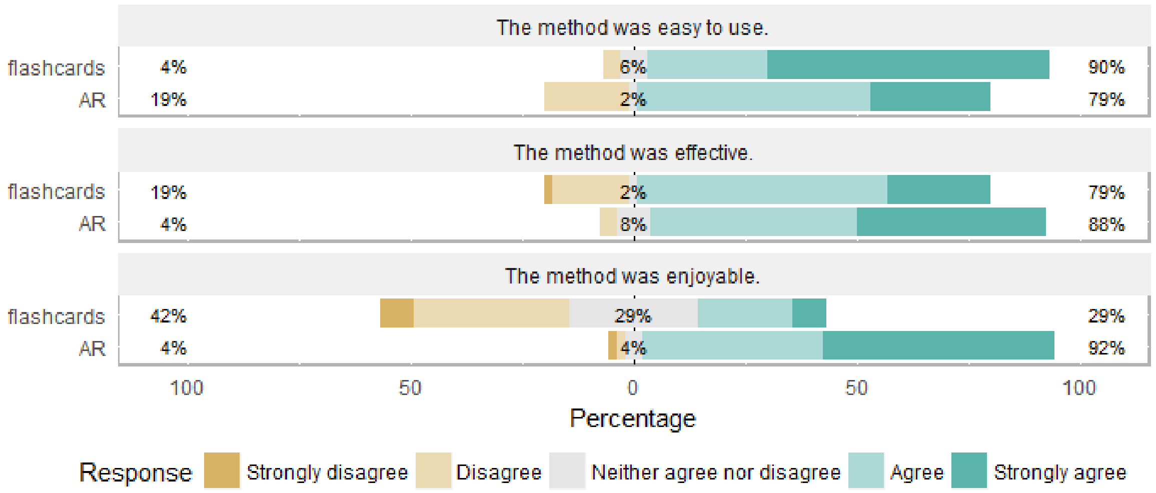


Figure 3.8: Qualitative feedback from the 52 participants

Figure 3.8 shows that when asked about the effectiveness of each platform for learning words, 88% of participants "somewhat agreed" or "strongly agreed" that the AR headset was effective, while 79% "somewhat agreed" or "strongly agreed" that the flashcards were effective.

Participants' comments comparing the two platforms revealed that about 20% (10 of 52) felt AR and flashcards were equally effective for learning because of the visual imagery both provide. 14 of 52 specifically mentioned that they found AR better because they saw the word and object at the same time. Almost 20% (10 of 52) stated that AR was better because it was more interactive, immersive, and showed objects in real time and space (e.g., *"The flashcards are classic and I have experience learning from them but the AR headset was more immersive"* and *"The headset was more interactive because it was right in front of you with physical objects rather than through a computer screen"*). Only 13% of the participants commented that flashcards were better, due to their familiarity with similar apps and computers in general.

A stark/striking difference was found in participants' opinions about which platform was enjoyable for learning. Figure 3.8 shows that 92% of participants "somewhat agreed" or "strongly agreed" that using the AR headset was enjoyable for learning words, compared to only 29% for using the flashcards. Open-ended comments from the participants pointed to the not unexpected novelty effect of AR (21 of 52 or 40%), *"The AR Headset because it was an incredibly futuristic experience."* In addition, 16 of 52 participants (31%) commented explicitly on how AR is more interactive, engaging, hands-on, natural, and allowed for physical movement (e.g., *"The AR headset was more interactive and required movement which engaged my mind more"* and *"The AR Headset was more fun because it's more fun to be able to move around and see things in actual space than on a computer screen"* or *"The AR headset was more enjoyable because it allowed for you to interact with the objects that you are learning about. It felt more realistic and applicable to real life, plus I had the visual image that helped me remember the words"*). Only 8 of 52 participants (15%) indicated that flashcards were more enjoyable because they were familiar, practical, and straightforward.

As we noted earlier in the discussion of productive recall results (Section 3.6.1, several participants commented in interviews or left text feedback related to the spatial aspect of the AR condition, generally saying that it helped give them an extra dimension to aid in learning. For example, one participant reported that: *"The AR headset put me in contact with the objects as well as had me move around to find words. I was able to recall what words meant by referencing their position in the room or proximity to other objects as well. ***Seeing the object at the same time as the word strengthened the association for me greatly***"*. Another participant said *"the AR seems like it would work better with friends or family trying to learn together, while the flashcards seem to work on an individual level."*. The latter comment points towards a social or interactive aspect of AR-based learning which we have not focused on in this study, but is nonetheless of potential interest to system designers and language learning researchers. The potential for social interaction and learning that this participant mentioned is likely linked to the availability of an interactive learning space.

Another possible benefit to learning in the AR condition is that it can facilitate the so called "method of loci" or "memory palace" technique [106]. It has been shown to be useful when applied to learn the vocabulary of a foreign language. The method is described for example in [107]. The author suggests to begin by creating a memory palace for each letter of the German alphabet by associating it with a location in an imaginary physical space. Each memory palace then is recommended to include a number of loci where an entry (a word or a phrase) can be stored and recalled whenever it is needed. One of our participants made a comment about this learning method after learning in the AR condition: *"I use memory palaces, so I really enjoyed AR as it felt somewhat familiar and made it easier for me to use the technique than the flashcards"*.

3.7 Limitations

Our proof-of-concept experiment shows that AR can produce better results on the learning of foreign-language nouns in a controlled lab-based user study. However, the study has several limitations. First, learning itself occurred in a controlled experimental context, in which subjects were paid an incentive. This cannot be assumed to be representative of real-world learning, and it is possible that our results may vary in real learning contexts. Second, and related, it is likely that novelty effects had some impact on the study given that the HoloLens remains in the category of new and exciting technology. Our design included a long acclimatisation phase with the device, but it is difficult to be sure that our qualitative results have not been impacted by novelty effects. Third, we chose to adopt a simple and standard implementation of the flashcard system, with words and pictures on opposite sides. This was fundamentally different from the AR condition, wherein labels and objects were visible at the same time. A small number of participants noted that they preferred the ability to view the object label and the object at the same time in the AR condition. On the other hand, others made use of the ability to self test in the flashcard condition. We are aware that our design choices and trade-offs will impact the learning experience. It is possible that other implementations would produce different results. Our results here represent a black-box comparison of these two learning approaches, and we encourage other researchers to extend our study to include other learning platforms and designs. Last, our productive recognition tests, while carefully controlled based on informal pre-studies and performance information from existing literature, showed ceiling effects with a large number of participants. No ceiling or floor effects were observed for the productive recall test. In follow-up experiments, we will increase the difficulty of the productive recognition tests.

3.8 Conclusions and Future Work

This chapter has described a 2x2 within-subjects experimental evaluation (N=52) to assess the effect of AR on learning of foreign-language nouns compared to a traditional flashcard approach. Key research questions were proposed, related to quantitative performance in immediate and delayed recall tests, and user experience with the learning modality (qualitative data). Results show that 1.) AR outperforms flashcards on productive recall tests administered same-day by 7% (Wilcoxon Signed-rank p=0.011), and this difference increases to 21% (p=0.001) in productive recall tests administered four days later. 2.) Participants reported that the AR learning experience was both more effective and more enjoyable than the flashcard approach.

These results are a good indication that AR can be beneficial for language learning, and I hope it inspires HCI and education researchers to conduct further research. AR language learning is a promising example application that appears to improve learning performance compared to other more traditional forms of technology assisted learning interfaces. Furthermore, I theorize that it achieves this benefit due to intrinsic properties that can only be achieved in AR. It only works because it sits in the cognitive feedback loop between your visual perception (what you see), your memory, and the learning process that happens over time. Ultimately, this is the kind of use case that we should be striving to achieve in our field. An AR system that doesn't simply grant the user a new suite of abilities, but empowers them to learn and retain new capabilities that persist even when the AR device is no longer in use. These are the kinds of next generation killer applications that I hope will drive the adoption of pervasive AR.

There are several avenues to continue research on the ARbis Pictus system, most notably, by taking the system beyond the controlled learning environment that was described in this paper and applying it to real-world learning tasks. However, there are numerous technical challenges to be solved before this vision of AR language learning, one that is always-on, with automatic personalization of the learning curriculum, and automatic generation of spatially arranged learning content, can come to fruition. To solve the personalized learning problem, we need to create a feedback mechanism for the system to determine how effective the current curriculum is and whether or not adjustments need to be made to better fit the user. In the next chapter of this dissertation, I will detail one possible avenue, eye tracking, and explore the feasibility of using this technology within the context of language learning. As an initial step towards tackling the content generation problem, I implement a first prototype of a real-time object labeling system with the HoloLens. In chapter 5 of this dissertation, I describe some preliminary evaluations of that system and how it can be incorporated into a language learning application.

While this dissertation largely focuses on technical solutions for AR language learning, there is also significant work to be done in the educational, psychological, and sociological domains. While personalized language learning plans exist, they have not been deployed at the temporal scale and with the degree of dynamicism that a pervasive AR system would demand. We need to involve tech savvy educators and course administrators to assist in the development of an appropriate language curriculum that can be deployed on ARbis Pictus. Furthermore, evaluating the performance of a real-world AR personalized-learning system is clearly a non-trivial task that will require complex longitudinal studies with many learners to account for differences in user experiences brought about by uncontrolled data in real-world environments. Ideally, we would lead a longitudinal study over the course of several weeks in a classroom, in order to measure the potential of AR in less controlled environments where the influence of novelty may be easier to measure and where students may interact with each others. In terms of education and learning theory, it may be possible for these results to expand the existing and established theories of CTML. It is also important to conduct further analysis and understand the differences of how various groups, such as multilingual people, approach vocabulary learning using this system.

Chapter 4

Using Eye Tracking to Measure Word Understanding

When designing pervasive AR applications, designers must think about all the various situations and use cases that a system might be deployed in, and adapt the application's feature sets accordingly. This can be a daunting task, and it can be difficult to know where to start. In this chapter, as well as the following chapter, I look at the context sensing and feedback loop, qualities of a pervasive AR system that enable increased application adaptability for everyday situations. Namely, I try to bridge the gap between the system's understanding of context and the user's experience of context, enabling new forms of context sensing capabilities to AR headsets.

Context is a multitude of things, and much like application design, it can be difficult to know where to start. In Grubert et. al's taxonomy of pervasive AR [2], the authors break down context sources into three types: Human, Environmental, and System factors. This is a helpful start, but doesn't really provide actionable goals for advancing the state of pervasive AR. In my work, I have filtered much of my exploration through the lens of enabling the AR language learning vision described in previous chapters. I have found it easier to narrow down research goals and scope projects by doing so, and the research described here is an outcome of that process.

Using Grubert et. al's taxonomy, we can describe this chapter as an exploration of human context sources, further filtered to those that are relevant to AR language learning. Learning is a cognitive process, so the human context we are most interested in is the learners cognitive state. There are many states that are valuable to the learning process. Interest, doubt, confusion, fatigue, could all be relevant when trying to infer the learners state of mind. However, the one that is most directly actionable is probably understanding. Whether or not a user recognizes or does not recognize something is indicative or whether they have retained the concept in their minds. If we could measure their level of understanding, we could compare their measured understanding to the application's expectations based on the current curriculum and have a quantifiable metric for how effective the application's teaching methods are.

Unfortunately, there are not many methods currently available for inspecting the users cognitive state. The most effective methods are often physiological and fairly invasive, something the average AR user is likely not willing to use. EEG for instance, requires careful placement of many electrodes on the users scalp, and often require the application of conductive gel to achieve reasonable accuracy. Eye tracking on the other hand, while not as accurate, has been correlated with a wide variety of cognitive states. Eye tracking hardware is also widely expected to be present in AR headsets, as researchers pursue foveated rendering to optimize rendering costs and increase performance on the small form factor, hardware constrained devices. In fact, the HoloLens 2, and certain VR headsets with AR functionality like the HTC Vive Pro, already have eye trackers embedded, though these devices were not yet available at the time of this work. For these reasons, we sought to explore the feasibility of using eye tracking data to determine word understanding. This work was done in collaboration with Jason Orlosky of Osaka University's CyberMedia Center, an expert in eye tracking for mixed reality. We conducted a user study in a VR environment to find and correlate eye tracking signals with word recognition, and the results of our exploratory research are detailed in the following sections.

4.1 Introduction

In recent years, augmented and virtual reality (AR/VR) have started to take a foothold in markets such as training and education. Although AR and VR have tremendous potential, current interfaces and applications are still limited in their ability to recognize context, user understanding, and intention, which can limit the options for customized individual user support and the ease of automation. AR has tremendous potential not only to add or modify content, but to enhance vision, memory, and even cognition. Quite a bit of literature exists on this topic, going back to the beginnings of AR [134, 135], but research on automatic assessment of user cognition is still limited in many ways.

One specific research area with great potential is that of learning enhancement. Lack of education across the globe is also still a significant problem. As a step towards improving education, our goal is to build an automated education framework that supports in-situ learning through AR and VR. As one step in this process, we need to better determine when an individual understands a particular concept and to what level. With respect to language learning, we need to recognize when a user remembers a particular word in a given context. To do so, we hypothesize that eye tracking can be used to classify a user's level of understanding of a particular event or concept when combined with context. In addition to observations of the tendencies of the eye during learning tasks, we evaluate a variety of different eye metrics to help with the classification of this kind of understanding.

More specifically, our system makes use of an eye tracked VR environment as a test bed. Using metrics including eye and head movement, pupillary response, and focus duration, we can to a certain degree classify the moment a user knows or is having trouble recalling a particular concept. We also hypothesize that we can determine the level of understanding or extent to which someone is able to recognize a particular concept based on the amplitudes and irregularities in some of these metrics. Most other work attempts classification of general cognitive activities over time, such as that by Henderson et al. or Marshall et al. [136, 137].

Our research differs from most prior studies in that we are evaluating the understanding of short-term, individual events as part of a specific context. Understanding events on a shorter time scale is important for learning interfaces since humans often learn new words or concepts in a matter of seconds.

Another contribution of this paper is the VR environment and series of experiments that will help benchmark suitable algorithms and reveal more about the physiological processes that occur during recall and understanding. Within our VR environment, we designed a series of word memory and tasks that should facilitate a certain amount of cognitive load. In comparison with previous studies that classify tasks based on viewing of images like that of Henderson et al. [136], we use an interactive environment to more closely resemble interactions with in-situ objects or tasks.

Results from our experiments show that fixation time, eye movement, and pupil size had the highest correlation to perceived word difficulty. Using these metrics, we were able to achieve a rate of 62.8% (known/recalled vs. unknown/forgotten) when trying to classify all easy, medium, and hard words, and 75.6% classification accuracy when considering only easy and medium difficulty words.

This work addresses the problem of automatically recognizing whether or not a user has an understanding of a certain term, which is directly applicable to AR/VR interfaces for language and concept learning. To do so, we first designed an interactive word recall task in VR that required non-native English speakers to assess their knowledge of English words, many of which were difficult or uncommon. Using an eye tracker integrated into the VR Display, we collected a variety of eye movement metrics that might correspond to the user's knowledge or memory of a particular word. Through experimentation, we show that both eye movement and pupil radius have a high correlation to user memory, and that several other metrics can also be used to help classify the state of word understanding. This allowed us to build a support vector machine (SVM) that can predict a user's knowledge with an accuracy of 62% in the general case and and 75% for easy versus medium words, which was tested using cross-fold validation. We discuss these results in the context of in-situ learning applications.

4.2 Eye Tracking

Eye tracking is a sensor technology that has a long history of use in both head-mounted display research and the cognitive sciences. In HMD research, accurate and robust eye tracking is the cornerstone of all foveated rendering techniques, which greatly improves rendering performance by reducing image quality outside of the current point of focus. The enormous potential of foveated rendering has led device manufacturers to preemptively include eye tracking cameras in the design of their headsets, such as with the HTC Vive Pro and the HoloLens 2, even when algorithms are not yet mature enough for use.

Within cognitive sciences, eye tracking has frequently found use as a non-invasive technique for measuring attention mechanisms or investigating the most central and provocative parts of a scene, otherwise known as saliency. But eye tracking is much more than just attention patterns. Eye tracking also provides auxiliary metrics such as pupil dilation and blink rate. Pupil dilation has been used to study mental workload, or the intensity and difficulty of cognitive processing. And blink rate has been used to study learning, working memory, decision making, and other aspects of cognitive development. Gaze can also be aggregated and analyzed over time, which can reveal cognitive strategies being employed in a given task. Eckstein et al. provide an excellent summary of the current uses of eye tracking in cognitive sciences literature [138].

Our investigation in this space revolves around the usability of eye tracking as a proxy for cognitive states in an AR application. Can we define a cognitive state that tracks closely with the goals of our application? How accurately can we measure it? What does that design process look like?

4.3 Methods

To answer these questions, we start by focusing on one of our representative applications: situated language learning. Situated language learning allows the user to utilize their environment as scaffolding in the learning process, by taking advantage of the brain's ability to efficiently link information that is highly cognizant and contextually relevant to the user, such as the familiarity of objects in their home. We have contributed some results in effectiveness of situated language learning, which can be found in previous chapters.

One of the primary challenges in language learning is determining and adapting to the rate of learning development, as every student is different. Some will learn faster than others, some will be more amenable to certain concepts, and some will simply have variations in discipline or motivation. In a classroom setting, it would typically be left to the teacher to determine the needs of each student and adapt the rate of material presented accordingly. The adaptation step can be relegated to personalized tutoring and curriculum software, but we still need humans-in-the-loop to determine the current state of language understanding. If we can directly measure the state of language understanding, we could create a fully automated interaction loop where educational content is dynamically adapted to the rate of learning and development. Coupled with the demonstrated efficacy of situated learning in AR, such a development would greatly alleviate the burdens of language teachers and make it easier for users to learn new languages.

The goal of this work is to investigate the feasibility of using eye tracking sensors on a head-mounted display to determine language understanding. While other papers have explored

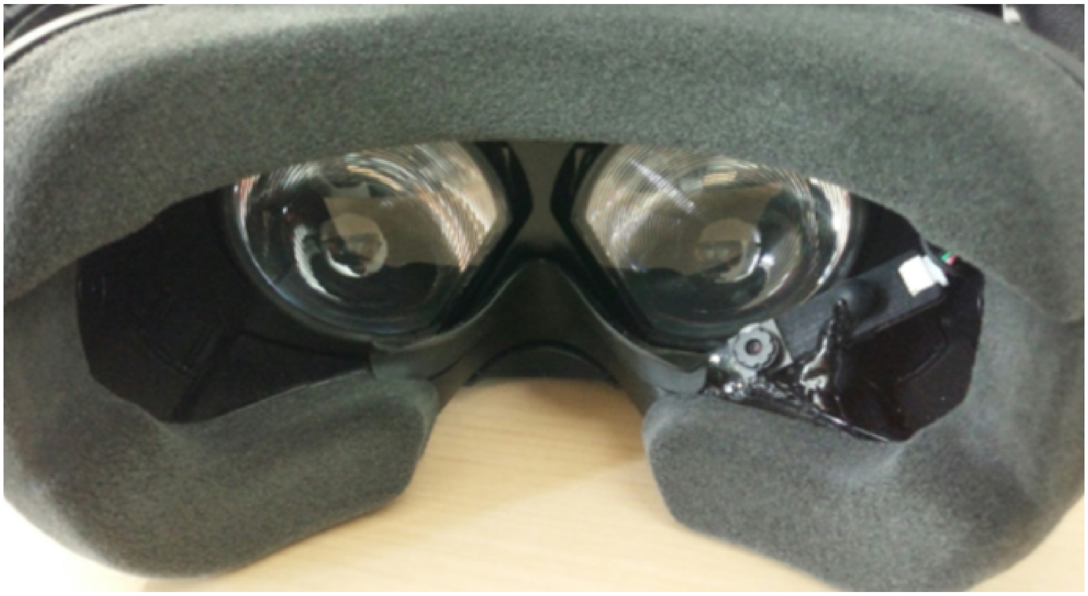


Figure 4.1: Eye tracking camera setup. Pupil Labs eye trackers were installed in an HTC Vive headset.

eye tracking as a signal for learning and cognitive development, our work is distinguished by its emphasis on the use of an HMD. Here, we opted to use an HTC Vive headset as our virtual reality apparatus, due to the lack of augmented reality headsets with eye tracking capabilities at the time of investigation. We installed Pupil Labs eye tracking cameras within the headset, as seen in Figure 4.1. Our eye tracking software is based on the same framework used by Itoh et. al., and makes use of a standard 5-point calibration procedure that was performed before the start of the task. After calibration, the user was asked to focus on a central point so that accuracy could be manually confirmed by the experimenter. If the calibration was off by more than two degrees, the calibration procedure was re-conducted.

Our virtual reality environment consisted of a room with six spawn points distributed equally in a circle around the participant. A visualization of this environment is shown in bird's eye view in Figure 4.2. English words were displayed within the centers of the spheres. Each sphere was spaced 1.2 meters away from the participant and had a radius of 0.5 meters, which defined a fixation point encompassing a field of view of approximately 23 degrees. Note

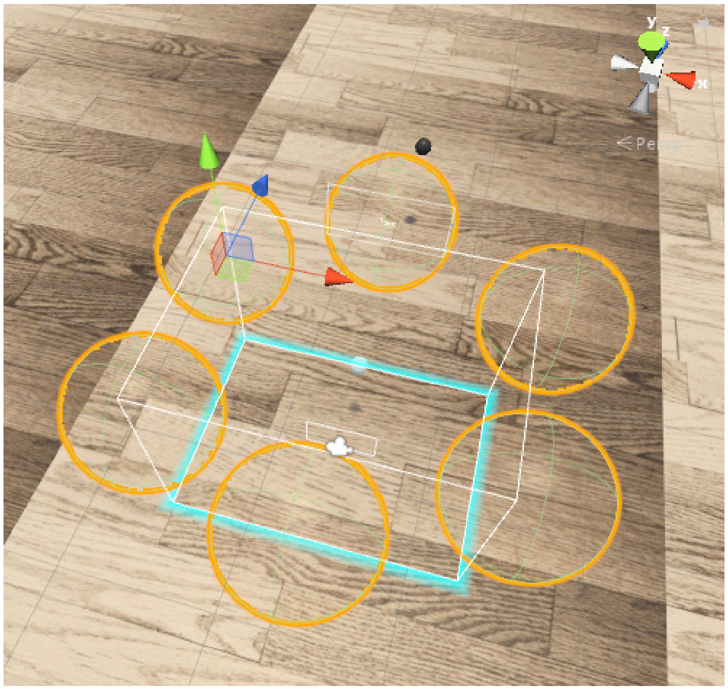


Figure 4.2: Virtual reality environment for the study.

that the sphere itself was not visible during the experiment, only the words. Each participant started at the center point equidistant to the spawn points during the tasks, though naturally that position drifted over time as they moved around to search for words during the task.

During the task, words were displayed randomly in one of the six spawn points. Participants were asked to read each word and answer yes or no, either by pulling the right or left trigger on the HTC Vive controllers, as shown in Figure 4.3. Subsequent words never overlapped on the same position as the previous word and always required participants to search and look around, ensuring that there is always a consistent gaze and fixation event for each word.

The words presented ranged in difficulty from very common words like "question" and "reason" to exceedingly rare words like "opsimath" and "cencacle". They were selected by randomly sampling words from a range of occurrence levels in the Google Ngrams database. These ranges were divided into "easy", "medium" and "hard" categories, defined by the occurrence ranges of 0.02% - 0.0011%, 0.001% - 0.0001%, and below 0.0001% occurrence,

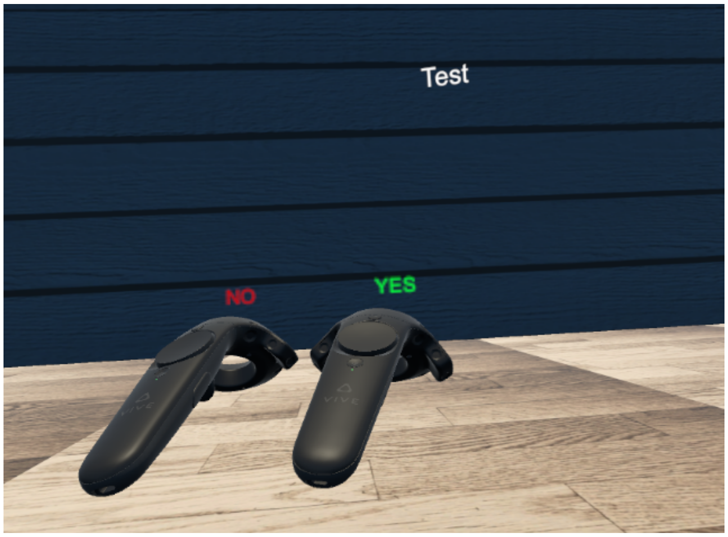


Figure 4.3: Task interface.

respectively.

The experiment was broken up into two tasks, 15 ordered words and 15 randomized words, picked randomly from the list of 30 words for each participant, while still ensuring there was even distribution of difficulties between the two tasks. The first 15 words were broken up into three groups, sequentially going from easy to medium to hard, with presentation of each of the 5 words in each group being randomized. The second set of 15 words was presented completely randomized with no difficulty ordering.

A total of 16 individuals (mean age of 31.8, stdev 8.16, range from 24 to 50) participated in the experiment. The participants came from a wide range of language backgrounds. All were non-native English speakers but had some English language ability, which also ensured a large distribution of answers. Moreover, any classification we achieve needs to be culture- and language- independent, so the larger variety of language abilities benefited our results. The experiment was conducted at Osaka University under IRB SA2016-2, and all participants signed a consent form prior to starting the experiment.

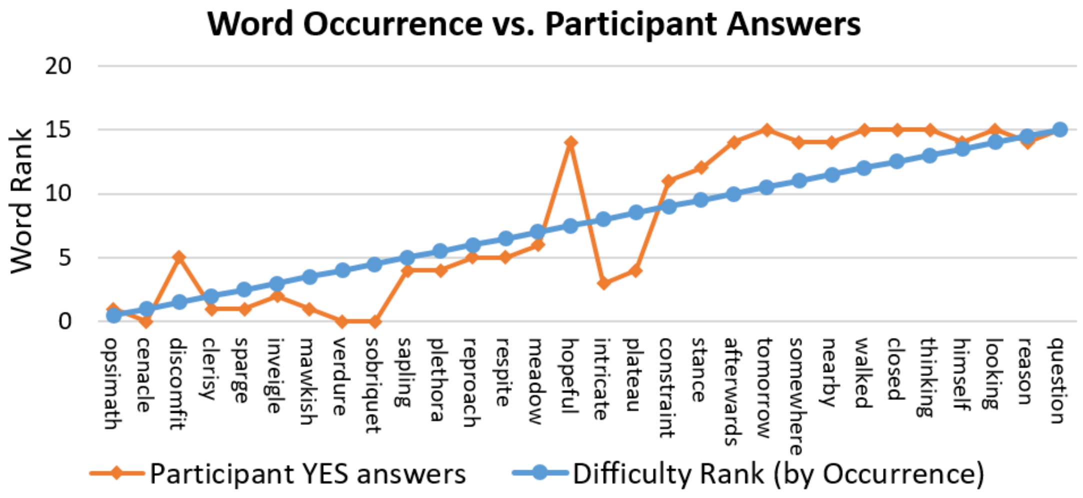


Figure 4.4: Word occurrence vs. Participant Answers.

4.4 Results

Due to the diverse backgrounds of the participants and variability in each individual's understanding of the English language, we first sought to verify the veracity of our rating of word difficulties. To test this, we mapped the words by Ngrams occurrence to the number of YES answers per participant, indicating how many participants in our study understood each word, shown in Figure 4.4.

As we can see from the graph, our method of ranking word difficulty by occurrence correlates well with participants' knowledge of the words. A statistical analysis further reveals a Pearson Correlation of $R(28) = 0.8983$ and $p < 0.01$.

4.4.1 Focus Time

Next, we wanted to determine whether the time spent focusing on a particular word differed by answer. The statistical analysis was conducted using a mixed effects model in R. For the binary outcome of YES or NO, we tested each of the metrics listed above for differences in means, while including participants as a random effect in the model. Within this model, type III

ANOVA using Satterthwaite's method and a separate Pearson correlation between each metric and word difficulty were computed.

First the average time spent responding to NO answers was 2753.69 ms vs. 1731.28 ms for YES answers. T-tests using Satterthwaite's method confirmed a significant difference in means for time $F(1,347.07) = 14.402$, $p < 0.001$. As a follow-up, we compared the time taken for answers in the ordered set of words versus the random set of words. This effect was not significant, $F(1,367.29) = 2.02$, $p < 0.155$, with the average times being 2396 ms for ordered answers and 2027 ms for random answers. Secondly, we wanted to see whether average time spent had a correlation to the number of known words and difficulty rank. Time and the number of YES answers for a particular word were not strongly correlated, with $R(28) = 0.291$, $p = 0.153$, and $R(28) = -0.351$, $p = 0.082$, for ranked difficulty. Though time data can help us classify binary understanding, it may not help establish the level of understanding of the word.

4.4.2 Head Movement

The next metric we explored was head movement. In particular, we analyzed head roll since several participants were observed cocking their heads to the side when thinking during the experiment. The average clockwise (from the participant's perspective) roll angle (abs value) was 3.61 degrees for NO answers versus 3.82 degrees for YES answers and counter-clockwise roll was 2.95 degrees for NO answers versus 2.81 degrees for YES answers. Neither of these were significant, with $F(1,352.21) = 0.550$, $p = 0.458$ and $F(1,328.16) = 0.013$, $p < 0.908$, respectively.

4.4.3 Eye Movement

The next and arguably most interesting set of metrics we used were those pertaining to eye movements. We analyzed saccades, blinks, and eye movement in pixels per second.

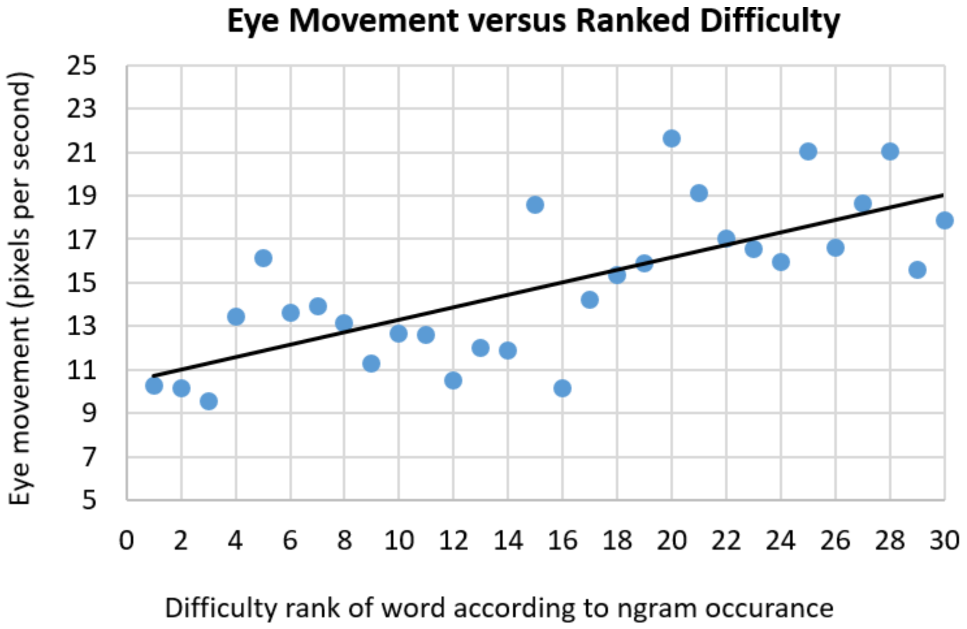


Figure 4.5: Graph showing the average eye movement in pixels per second against word difficulty ordered by increasing difficulty, or ranked difficulty. Pearson correlation with $R = .8983$, $P < 0.01$.

First off, neither saccades nor blink frequency were found to be significant. These were measured by dividing the total number of occurrences over fixation time to establish frequency. No effect was found for saccades, with frequency for YES as 1.99 saccades/sec vs 1.97 saccades/sec for NO, $F(1,263.39) = 0.412$, $p = 0.522$. Moreover, Pearson correlation was equal to $R = 0.056$, $p = 0.882$, which was not significant. Blink frequency was also non-significant, with YES as 0.211 blinks/sec vs 0.221 blinks/sec for NO, $F(1,174.29) = 0.06$, $p = 0.807$. Pearson correlation was equal to $R = 0.096$, $p = 0.743$.

The most significant metric turned out to be eye movement, i.e., the average movement per second from the time the participant began gazing at the word from the time they moved on to the next word. Results are separated into X, Y, and total Euclidean distances. Average magnitude of X velocity (in pixels per second) was 9.276 for NO and 14.422 for YES, for which the difference was significant, $F(1,362.39) = 43.41$, $p < 0.0001$. For Y velocity, this was 7.28 for NO answers versus 10.274 for YES answers, for which the difference was also significant

$F(1,353.89) = 15.098$, $p < 0.0001$. Total Euclidean distance was 12.048 for NO answers versus 18.086 for YES answers, for which the difference was also significant $F(1,365.04) = 38.591$, $p < 0.0001$.

Moreover, all movement, including the Euclidean distance from the previous X,Y position to the next yielded significant Pearson Correlations for:

- X mvt. vs total YES answers: $R(28) = 0.798$, $p < 0.01$
- Y mvt. vs total YES answers: $R(28) = 0.693$, $p < 0.01$
- X mvt. vs difficulty rank: $R(28) = 0.674$, $p < 0.01$
- Y mvt. vs difficulty rank: $R(28) = 0.718$, $p < 0.01$
- Euclidean total of mvt. vs answers: $R(28) = 0.801$, $p < 0.01$
- Euclidean total of mvt. vs rank: $R(28) = 0.724$, $p < 0.01$

A graph of euclidean total eye movement in pixels per second vs. ranked difficulty of words is shown in Figure 4.5, where the correlation can be clearly visualized.

4.4.4 Pupillometry

Finally, we looked at metrics related to the pupil, namely pupil size and pupil deviation. Some pupillometric measures were significant between answers.

Average absolute pupil radius for YES was 2.18 mm versus 1.92 for NO, $F(1,375.59) = 2.643$, $p = 0.105$. Pupil radius was well correlated to ranked difficulty, $R(28) = 0.634$, $p < 0.01$, and to answers, $R(28) = 0.720$, $p < 0.01$, which is also shown in Figure 4.6. Pupil deviation was also very significant, resulting in YES answers at 0.478 mm/sec and NO answers at 0.371 mm/sec, which represent the magnitude of any changes in pupil size, $F(1,376.98) = 27.42$, $p < 0.0001$.

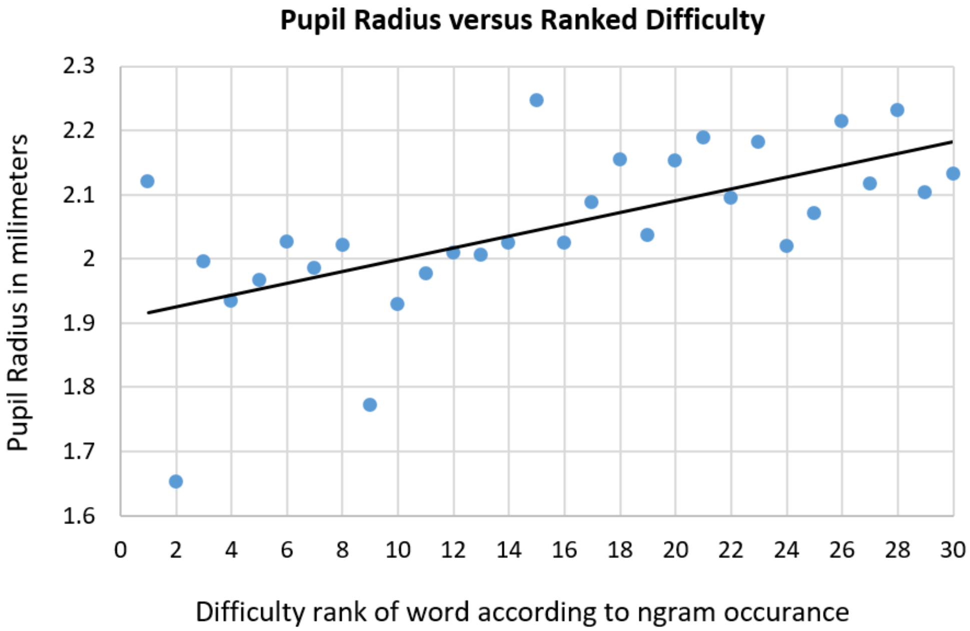


Figure 4.6: Graph showing the average pupil radius in millimeters versus ranked difficulty.. Pearson correlation of $R(28) = 0.720$, $p < 0.01$.

Table	**User Selections**		
Metric	***YES***	***NO***	***Unit***
Fixation Time	1.73	2.75	seconds
Eye Movement (Eucl.)	18.086	12.048	pixels/second
Pupil Deviation	0.487	0.371	millimeters/second
Absolute Pupil Size	2.18	1.92	millimeters

Table 4.1: Summary of all statistically significant data to be used in the SVM classifier.

4.5 Learning State Classification

The most significant of these data are summarized in Table 4.1 below. All of these metrics were then used to train a Support Vector Machine (SVM) for classification of Understood (YES) vs. Not-understood (NO) words.

For the initial SVM design, we used a single class, supervised linear SVM from the Shark library [139]. We first used all of the available data points (374 user selections) as input to the SVM and ran a full cross-validation. This resulted in a subject-agnostic model with classification accuracy of 62.8% (235 / 374) for any user. Note that a small number of outliers (16) were removed due to issues with the eye tracker disconnecting during the study.

However, further inspection showed that hard words appeared to be more difficult to classify than easy or medium words. We believe this is because hard words tend to have a lower fixation time than medium words, causing the SVM to mix up easy and hard responses.

As such, we re-ran the data to try and classify YES/NO answers for just the subset of words containing easy and medium difficulties. As we hypothesized, the SVM classification improved to 75.6% (198 / 262 correct). This accuracy will likely be even higher with a personalized classification model, improved eye tracking, and additional training data.

4.6 Applications and Use Cases

Despite the existing body of research on cognitive state recognition, we still lack concrete ways to modulate or present virtual content based on the resulting output, especially for short-term events. For example, many systems can determine that a user is confused or engaged in visual search over a longer period of time, but very few researchers have focused on how to overlay instructions or augmentations in response to those mental states, let alone the environment. With this project, we have demonstrated a new capability — whether or not the user recognizes and understands the content they are interacting with.

As mentioned in the beginning of this chapter, this study was motivated by the desire to improve the capabilities of language learning interfaces in AR. This cognitive state detection fits into a larger system, along with a language model and intelligent tutoring system, enabling an automated recognition system that can provide a more intelligent, in-situ, natural way of learning through AR. The theoretical design and implications of such a system is described in chapter 5 of this dissertation. Here, we simply highlight the utility of cognitive state detection via head-mounted eye tracking.

One other potential example is that of assistive technologies for the elderly and disabled. Consider an elderly user who sometimes forgets to take their medication, follow procedural instructions, or interact with an object of significance. To recall an appropriate augmentation for that object at the right time, we need to understand 1) the user's context, 2) the state of that object within its context, and 3) the user's mental state in relation to that object and task. In other words, we need to determine whether a user would say yes or no to questions such as "do you understand where you are," "do you understand this word," or "are you confused?" Being able to classify a binary yes or no to these questions can help a ubiquitous computing system determine how to display the appropriate navigation interface, word learning annotation, or medication checklist, or signal an external party for help. We could train a model similar to the one presented here for cognitive states and learning events related to conditions like dementia or memory loss. These cognitive markers could then be associated with an external database to assist the user in recovering relevant information, effectively extending their human memory capacity.

4.7 Discussion

We have demonstrated the feasibility of detecting learning states in virtual reality. We also identified features that correlate closely to learning states. We believe that there is a significant degree of transferability to augmented reality settings. We discovered several statistically significant eye camera signals and were able to train a simple classifier using these data points to predict word understanding on our ground truth dataset with reasonable accuracy.

In the experiment, we were able to achieve between 62.8% and 75.6% accuracy for the user's understanding of a short-term word recognition task. However, our SVM classifier utilizes aggregated values within a certain window of time: between when the word enters their field of view up to the point they respond. At the time a user infers meaning in a practical situation, the fixation duration is not as easy to delineate, which may affect the accuracy of classification. Creating a personalized model for each user could help alleviate bad classifications. Coupled with future improvements, for instance increased eye camera resolution, and better machine learning classifiers, such as attention based neural networks, we expect the accuracy of this method to improve substantially.

One other source of error from the experiment could be our detection algorithms for blinks and saccades. Several other APIs exist in addition to our custom built detectors, however we do not have a way to benchmark these algorithms against ground truth. Updated eye tracking hardware may alleviate this potential source of error in the future. Moreover, better eye tracking algorithms will also likely reduce the error in calculated pupil size, further improving classification accuracy.

One interesting and somewhat counter-intuitive result from our experiments was that medium difficulty words were more easily separable from easy words than were hard words. Through observation and post-experiment discussion with participants, we concluded that it is easy to know when a person doesn't know a word at all since there is no stored memory of the word to recall. Conversely, when a word of medium difficulty is unknown and the participant has either known it and forgotten or had some visual or aural exposure to the word, he or she will have to access his or her memory in more depth and expend more cognitive energy to determine whether or not the meaning is known. This contrasts somewhat with results found by Karolus et al., where increased fixation time was correlated with lower language ability [140]. As such, context (for example reading versus recalling an individual word) seems to play an extremely important role when deciding what data to use for which classifier.

Chapter 5

Using Object Recognition for Content Management

Previous chapters introduced a conceptual AR language learning system as a framework for exploring challenges in the design of pervasive augmented reality applications and investigated the use of eye tracking as part of the larger envisioned system to utilize user context for learning in augmented reality. Those results contributed towards solving the problem of personalized learning adaptation, or how a pervasive AR application could adapt to human context information. But we have yet to touch on environmental context however. Using the lens of our AR language learning application, we can identify one major bottleneck in the creation of learning content.

Generating new content is trivial, as it simply involves looking up the next stage of progress in the learning curriculum. But that alone would not make for a unique AR application with features distinct from other computing mediums. We want to take advantage of spatial arrangement to improve learning and recall, as described in previous chapters. In other words, we need a way to automatically and seamless arrange learning content within any physical environment we encounter. To do so requires context awareness in the form of semantics and geographical positioning. Thankfully, the deployment of deep learning in Computer Vision tasks has drastically improved the accuracy of recognition, tracking, and localization algorithms for a variety of context levels. Unfortunately, these deep learning models typically have billions of parameters and require a powerful GPU or multiple GPUs to run. Current optical see through headsets just aren't capable of running these models natively.

This chapter tackles the challenge of how to enable automatic content arrangement and adaptation to new physical environments. We introduce new context awareness capabilities for augmented reality in the form of object recognition, specifically demonstrating that it is possible to use state of the art object recognition models on a per-frame basis through over-the-network video streaming to a local GPU server. Our implementation enables the usage of large vision models without sacrificing portability or needing to wear bulkier hardware. We evaluate real-world performance of the implementation in a small pilot study.

Additionally, this chapter discusses the entire process of designing this language learning system, including background and motivations, technical challenges, system architecture and characteristics, and lessons learned. We detail which assumptions were made about user context and environmental context for the system. We also describe and analyze the choices made in terms of algorithm selection, including the choice of deep learning model. The design of this system presents insights into the challenges of integrating multiple disparate components, such as eye tracking and object recognition, which operate at different throughput, into a resource-constrained mobile augmented reality headset. By undertaking the design and feasibility evaluation process, we contribute insight into how to incorporate state-of-the-art techniques such as deep learning and identify roadblocks towards future pervasive AR application implementations.

5.1 Motivation

For many years, learning new words has often been accomplished by memorization techniques such as flash cards and phone or tablet based applications. These often use temporal spacing algorithms to modulate word presentation frequency such as Anki [141] and Duolingo [142]. A more effective, albeit time consuming, method of language learning is to attach notes with words and illustrated concepts to real world objects in a familiar physical space, taking advantage of the learner's capacity for spatial memory. Learners constantly see a particular object, recall the associated word and learn that concept more effectively since the object is in its natural context and is consistently viewed over time. This type of learning is also referred to as the method of loci [106, 107, 143].

Our goal is to replicate this in-situ learning process, but to do so automatically and with the support of augmented reality (AR), as represented in Fig. 5.1 b. In other words, when a user views an object, we want to automatically display the concept(s) associated with that object in the target language and provide a method for both the viewing and selection of a particular term or concept. Deploying such an interface in a real-world, generalized context is still a very challenging task.

As a step towards this goal, we introduce a more practical framework that can function as a cornerstone for improving in-situ learning paradigms. In addition to the process of trial and error to find a more effective and practical approach to designing such a system, our contributions include:

1. a client-server architecture that allows for real-time labelling of objects in an AR device (Microsoft HoloLens),

2. a description and solution to the object registration problem resulting from the use of real-time object detectors (Fig. 5.1 a),

3. a practical framework for exploring challenges in the implementation of AR language learning, and a discussion of novel interaction paradigms that our framework enables.

The practical use of this system can enable in-situ learning for languages, physical phenomena, and other new concepts.

5.2 Related Work

Prior work falls into three primary categories, 1) the implementation of object recognition, semantic modeling, and tracking for in-situ labeling, 2) view management techniques for labeling in AR, and 3) the use of AR and VR to facilitate learning of concepts and language. While all of these three categories are typically different areas of research, they are each essential for the effective implementation of in-situ AR language learning.

5.2.1 Object Recognition and Semantic Modeling

Real-time object detection is a fairly new development, and there are not many works discussing the integration of these technologies into an augmented reality system. Current detection approaches utilize object recognition in 2D image frames, using learning representations such as Deep and Hierarchical CNNs and Fully-Connected Conditional Random Fields [144, 145], or, for fastest real-time evaluation performance just a single neural network applied to the entire image frame [146]. Combined 2D/3D approaches [147, 148] or object detection in 3D point cloud space [149, 150] may become increasingly feasible for real-time approaches in the not-too-far future as more 3D datasets [148, 149] become available, but currently, approaches that apply 2D object detection to the 3D meshes generated by AR devices such as HoloLens or MagicLeap One yield better performance.

Huang et al. [33] compare the general performance of three popular meta architectures for real-time object detection. They show that the Single Shot Detector (SSD) family of detectors, which predicts class and bounding boxes directly from image features, has the best performance to accuracy tradeoff. This is compared to approaches which predict bounding box proposals first (Faster-RCNN and R-FCN). We experimented with the performance of both types of detectors and ultimately settled on an implementation of SSD.

The most recent and closest work to our approach is that of Runz et al. [32] in 2018. Using machine learning and an RGBD camera, they were able to segment the 3D shapes of certain objects in real time for use in AR applications. Their approach utilized the Mask-RCNN architecture to predict per-pixel object labels, which comes at a higher performance cost. In contrast, our approach is implemented directly on an optical see-through HMD (HoloLens) using a client-server architecture, and uses traditional bounding box detectors which can run in true real time (30fps) with few dropped frames.

Our work links objects that are recognized in real time in 2D frames to positions in the modeled 3D scene, which is akin to projecting and disambiguating 2D hand-drawn annotations into 3D scene space [151].

5.2.2 View Management for Object Labeling

A body of work in AR research focuses on optimized label placement and appearance modulation. In a similar fashion that we use 2D bounding boxes of recognized objects in the image plane to determine a 3D label position for that object, several view management approaches optimize the placement of annotations based on the 2D rectangular extent of 3D objects in the image plane [152–154]. Other approaches allow the adjustment of labels in 3D space [155, 156], a feature that might be gainfully employed in our system to subtly optimize the location of an initially placed label over time as multiple vantage points accumulate. However, this would pose the additional problem of disruptive label movement, due to loss of temporal coherence.

Since potential mislabeling actions due to occlusions – the main motivation for 3D label adjustment – are automatically resolved by the HoloLens' continuous scene modeling (occluders are automatically modeled as occluding phantom objects), we can simply avoid label adjustment after we arrived at a good initial placement. Label appearance optimization [157] and assurance of legibility [158, 159] are beyond the scope of this paper.

5.2.3 Memory and Learning Interfaces

The idea of augmenting human memory or facilitating learning with computers appeared almost simultaneously with the history of modern computing. For example, early work by Siklossy in 1968 proposed the idea of natural language learning using a computer [160]. Since then, much progress has been made, for example by turning the learning process into a serious game [161]. Though not in an in-situ environment, Liu et al. proposed the use of 2D barcodes for supporting English learning. Though relatively simple, this method helps motivate the use of AR for learning new concepts, as a form of fully contextualized learning [162].

In addition to language learning, some work has been presented that seeks to augment or improve memory in general. For example, the infrastructure proposed by Chang et al. facilitated adaptive learning using mobile phones in outdoor environments [163]. Similarly, Orlosky et al. proposed the use of a system that recorded the location of objects, such as words in books, based on eye gaze, with the purpose of improving access to forgotten items or words [164].

Other studies like that of Dunleavy et al. found that learning in AR is engaging, but still faces a number of technical and cognitive challenges [165]. Kukulska-Hulme et al. further reviewed the affordances of mobile learning, having similar findings that AR was engaging and fun for the purpose of education, but found that technology limitations like tracking accuracy interfered with learning [166]. One more attempt at facilitating language learning by Santos et al. used a marker based approach on a tablet and tested vocabulary acquisition with marker-based AR. In contrast, our approach is designed to be automatic, and is a hands-free in-situ approach.

Most recently, Ibrahim et al. examined how well in-situ AR can function as a language learning tool [70]. They studied in-situ object labelling in comparison to a traditional flash card learning approach, and found that those who used AR remembered more words after a four day delayed post-test. However, this method was set up manually in terms of the object labels. In other words, the objects needed to be labelled manually for use with the display in real time. In order to use the display for learning in practice, these labels need to be placed automatically, without manual interaction.

This is the main problem our paper tackles. We have developed the framework necessary to perform this recognition, and at the same time we solve problems like object jitter due to improper bounding boxes. This sets the stage for a more effective implementation of learning via the method of loci, and can even enable reinforcement type schemes like spacing algorithms [141] that adapt to the pace of the user based on real world learning.

5.3 AR Language Learning Framework

As further motivation for this system, we envision a future where Augmented Reality headsets are smaller and more ubiquitous, and are capable of being worn and used on a daily basis much like current smart phones and smart watches. In such an “always-on AR” future, augmented reality has the potential to transform language learning by adapting educational material to the user’s own environment, which may improve learning and recall. Learning content may also be presented throughout the day, providing spontaneous learning moments that are more memorable by taking advantage of unique experiences or environmental conditions. Furthermore, an always-on AR device allows us to take into consideration the cognitive state of the user through emerging technologies for vitals sensing. Using this information, we can

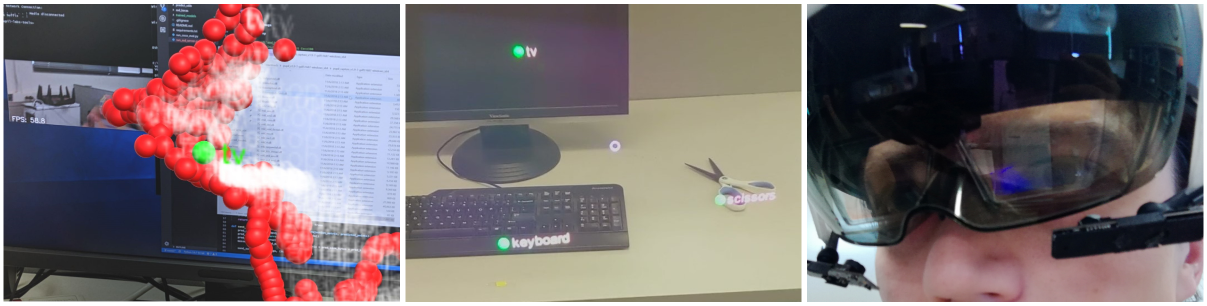


Figure 5.1: Images showing a) our object registration algorithm, which uses a set of uncertain candidate object positions (in red) to establish consistent labels (in green) of items in the real world b) a view directly through the HoloLens of resulting labels from our method in a previously unknown environment, and c) a photo of a user wearing the system and calibrated eye tracker used for label selection.

gain a better understanding of the user's attention, and more readily adapt to their needs. To enable research into these interaction paradigms, we propose a practical framework that can be implemented and deployed on current hardware using current sensing techniques. We believe the fundamental building blocks for AR language learning include three components:

- Environment sensing with object level semantics
- Attention-aware interaction
- Personalized learning models

These components provide the necessary set of capabilities required by the AR language learning applications we envision. In the next section, we will introduce a system design which implements this framework using existing technologies. Then, we will describe the realization of the first component of our framework, through an object level semantic labeling system. Finally, we will discuss our ongoing work regarding the second and third components.

5.4 System Design

In this section, we introduce a client-server architecture composed of several interconnected components, including the hardware used for AR and eye tracking, the object recognition system, the gaze tracking system, and the language learning and reinforcement model. The overall design and information flow between these pieces and parts is shown in Figure 5.2.

The combination of these pieces and parts allow us to detect new objects, robustly localize them in 3D despite jitter, shaking, and occlusion, and label the objects properly despite improper detection. Our current implementation targets English as a Second Language (ESL) students, thus our labels are presented in English. But the label concepts could be translated and adapted to many other languages.

5.4.1 Hardware

We chose the Microsoft Hololens for our display, primarily because it provides access to the 3D structure of the environment and can stream the 2D camera image to a server for object recognition. How we project, synchronize, and preserve the 2D recognition points onto their 3D positions in the world will be described later.

The HoloLens is also equipped with a 3D printed mount that houses two Pupil-Labs infrared (IR) eye tracking cameras, as shown in Fig. 5.1 c). These cameras are each equipped with two IR LEDs, and have adjustable arms that allow us to adjust the camera positions for individual users. The eye tracking framework employs a novel drift correction algorithm that can account for shifts on the user's face.

For the server side of our interface, we utilized a VR backpack with an Intel Core i7-7820HK and Nvidia Geforce GTX 1070 graphics card. Since the backpack is designed for mobile use, this allows both the Hololens and Server to be mobile, as long as they are connected via network. To maximize throughput during testing and experimentation, we connected both

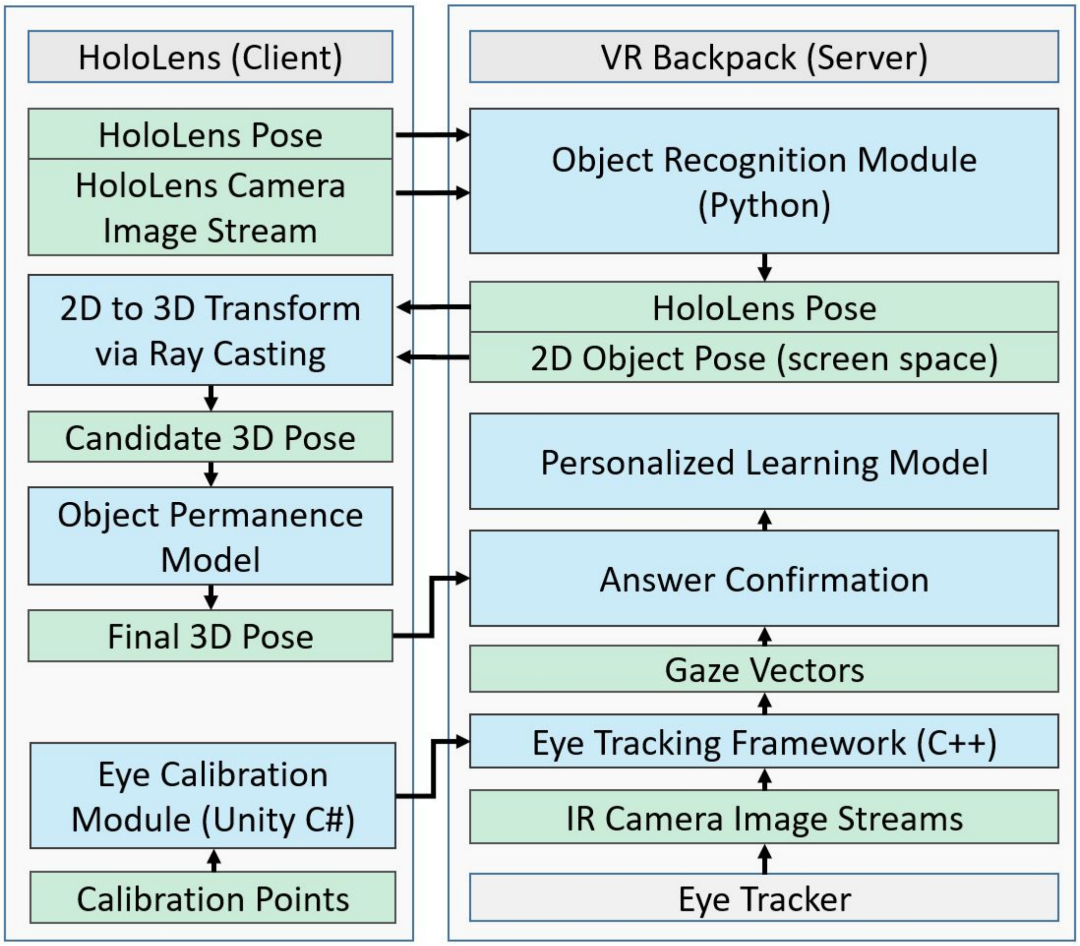

HoloLens (Client)
HoloLens Pose
HoloLens Camera Image Stream
2D to 3D Transform via Ray Casting
Candidate 3D Pose
Object Permanence Model
Final 3D Pose
Eye Calibration Module (Unity C#)
Calibration Points
VR Backpack (Server)
Object Recognition Module (Python)
HoloLens Pose
2D Object Pose (screen space)
Personalized Learning Model
Answer Confirmation
Gaze Vectors
Eye Tracking Framework (C++)
IR Camera Image Streams
Eye Tracker

Figure 5.2: Diagram of our entire architecture, including hardware in grey, algorithms and systems in blue, and data flow in green. The left-hand block includes all processing done on the Hololens and the right-hand block includes all processing done on the VR backpack.

devices on the same subnet.

5.4.2 Summary of Data Flow

Our system starts by initializing the Unity world to the same tracking space as the Hololens. Next, we begin streaming images from the Hololens' forward-facing camera, which are sent to and from the server-side backpack via custom encoding. Upon reaching the server, they are decoded and input into the object recognition module, which returns a of 2D bounding box with an object label. The center of this bounding box is then sent back to the Hololens and projected into 3D world space by raycasting against the mesh provided by the Hololens. This projected point is treated as a “candidate point”, which is fed into our object registration algorithm. The object registration algorithm looks over the set of candidate points over time to decide where to assign a final object label and position. Once an object and its position have been correctly assigned, the object is synchronized with the Unity space on the server side. Finally, labels on the objects are activated using eye-gaze selection, giving the user a method for interaction. The results from this interaction are fed into a personalized learning model, providing the ability to design content that adapts to the growth of the user.

5.5 In-Situ Labeling

The success of Convolutional Neural Networks (CNNs) has lead to technological breakthroughs in object recognition. However, it is not yet obvious how to integrate these technologies into AR. Three major parts need to be in place for these tools to be used practically. First, they need to be tested in practice (not just on individual image data sets) and provide good enough recognition to label an object correctly over time. Secondly, we need to establish object registration that is resilient to failed recognition frames, jitter, radical changes to display orientation, and objects entering/leaving the display’s field of view (FoV). Finally, current AR devices are not powerful enough to run state-of-the-art CNNs. We need to handle the synchronization and reprojection between streamed frames from the AR device and recognition results from a server with a powerful GPU.

5.5.1 Object Recognition Module

The first step for the development of our system was finding a scalable object recognition approach that could be used with the forward facing camera on the HoloLens. Due to the real-time performance constraint, we had to test and refine a variety of approaches before finding one that worked. We finally found the Single Shot MultiBox Detector (SSD) by Liu et al. to be effective [167]. Specifically, we use the implementation provided by the TensorFlow Object Detection API, using the ssd_mobilenet_v1_coco model, which has been pre-trained on MS COCO.

We stream video frames from the built-in HoloLens front facing camera to a server running on an MSI VR backpack. To keep packet sizes small, we used the lowest available camera resolution of 896x504. Each frame is encoded into JPEG at 50% quality, so that their final size fits into a single UDP packet. We also encode and send the current camera pose along with each frame. On the server side, we place all frames into an input queue. An asynchronous processing thread takes the most recent frame from the input queue and feeds it through the SSD network. The resulting 2D bounding boxes and labels are then sent back to the HoloLens, along with the original camera pose. Back on the HoloLens, we project the center point of each 2D bounding box onto the 3D mesh by performing a raycast from the original camera pose.

This particular implementation of SSD takes 30ms per prediction on the VR backpack, which just barely allows us to achieve 30fps under ideal network conditions. There is a slight delay due to network latency, as our network has a round trip time of 150ms.

SSD and similar CNN based real-time object recognition architectures are known to perform poorly with small objects [33]. In practice, we found that small objects, such as spoons and forks, experience much higher false positive rates and predictions are not consistent across frames. Large objects are more reliable, such as predictions for TVs, chairs, and people. For medium sized objects, typically performance improves under realistic environmental conditions where the camera is able to capture more contextual information, such as keyboards and mouses being near each other.

To solve this problem, we make use of multiple streamed frames to establish an initial estimate of the object's location, confirm this location using a sliding window approach based on past labels and proximity, and finally assign a position for the label. This results in a very

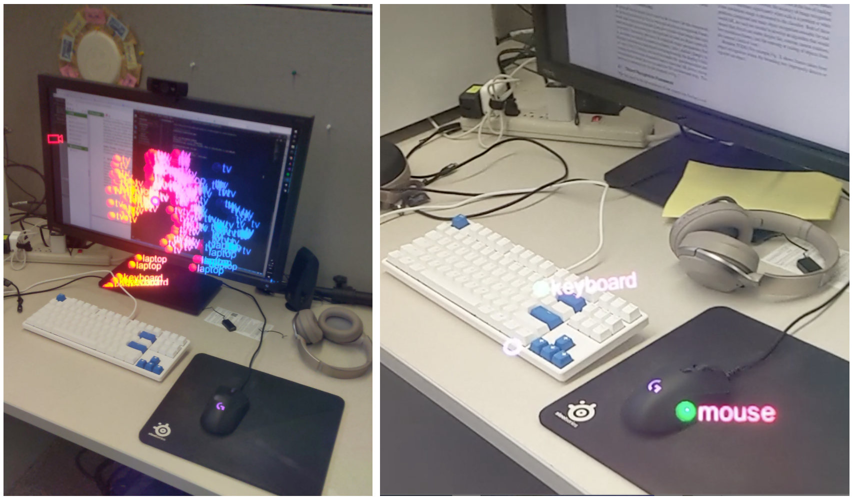


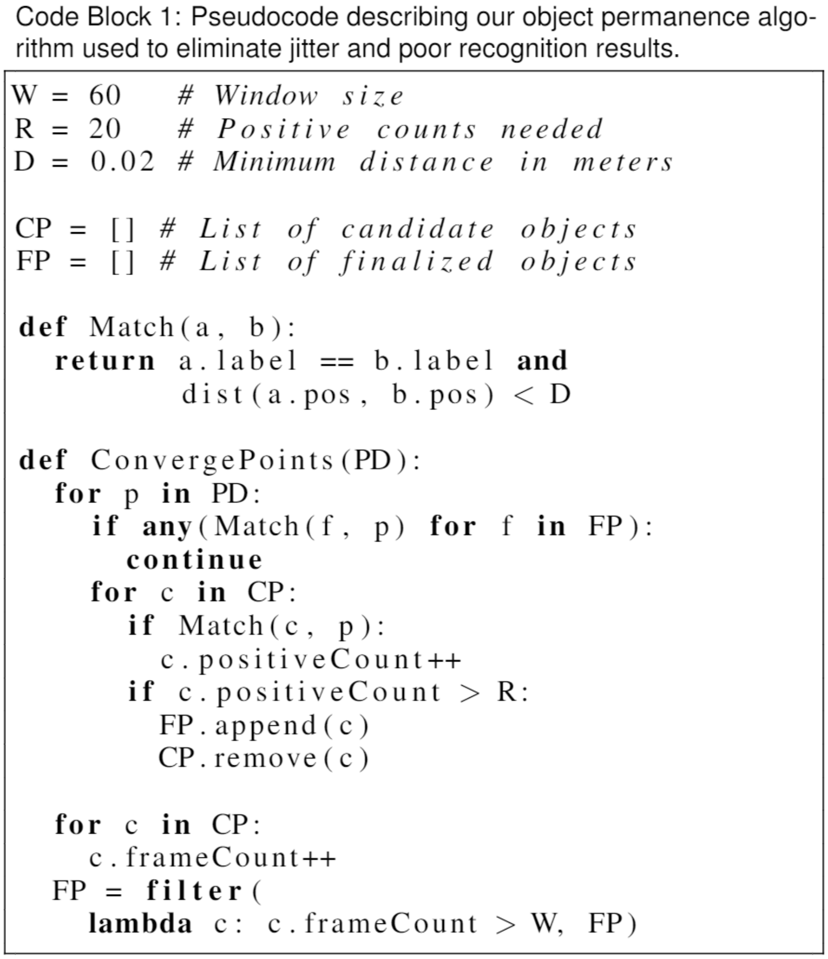
Figure 5.3: Left: Raw points returned from object recognition as projected into 3D space, accumulated over several frames. This shows the variance in predicted positions and false positive label predictions. Right: Scene correctly labeled with object-permanent labels.

stable, properly registered augmentation that is persistent despite various camera rotations or traveling in and out of various areas of a workspace. The algorithm we use for this purpose is described as follows:

First, an image streamed from the forward-facing HoloLens camera is passed to the SSD network, which then provides an initial prediction for a given object location in the form of a 2D bounding box. This 2D pose (i.e. the center of the bounding box in screen space) is then sent back to the HoloLens, and it is projected into 3D space as summarized previously.

Second, for every subsequent prediction, we check every instance of the same label in 3D space for the past W frames. A grouping of some of these labels can be seen on the left of Fig. 5.3. If the Euclidean distance between these subsequent 3D positions are within a threshold D (e.g. 50 centimeters away for a keyboard object), we average these positions and affix the object. After thorough testing and refinement, we found that object predictions converge well if there are 20 positively identified instances over a window of $W = 60$ frames under the defined threshold. An example of successful assignment of objects can be seen on the right of Fig. 5.3.

One advantage of this approach is that we can use semantic information to help guide the



Code Block 1: Pseudocode describing our object permanence algorithm used to eliminate jitter and poor recognition results.

```
W = 60    # Window size
R = 20    # Positive counts needed
D = 0.02 # Minimum distance in meters

CP = [] # List of candidate objects
FP = [] # List of finalized objects

def Match(a, b):
  return a.label == b.label and
         dist(a.pos, b.pos) < D

def ConvergePoints(PD):
  for p in PD:
    if any(Match(f, p) for f in FP):
      continue
    for c in CP:
      if Match(c, p):
        c.positiveCount++
      if c.positiveCount > R:
        FP.append(c)
        CP.remove(c)

  for c in CP:
    c.frameCount++
  FP = filter(
    lambda c: c.frameCount > W, FP)
```

Figure 5.4: Pseudo-code describing our object permanence algorithm used to eliminate jitter and poor network results.

distance threshold. For example, a sofa might use points spaced one meter away versus a pencil with points less than ten centimeters away.

First, an image streamed from the forward-facing HoloLens camera is passed to our object recognition system, which then provides an initial prediction for a given object location in the form of a 2D bounding box. This 2D pose (i.e. the center of the bounding box in screen space) is then sent back to the HoloLens, and it is projected into 3D space as summarized previously.

Second, for every subsequent prediction, we check every instance of the same label in 3D space for the past W frames. A grouping of some of these labels can be seen on the left of Fig.

5.3. If the Euclidean distance between these subsequent 3D positions are within a threshold D (e.g. 50 centimeters away for a keyboard object), we average these positions and affix the object. After thorough testing and refinement, we found that object predictions converge well if there are $R = 20$ positively identified instances over a window of $W = 60$ frames under the defined threshold.

One advantage of this approach is that we can use semantic information to help guide the distance threshold. For example, a sofa might use points spaced one meter away versus a pencil with points less than ten centimeters away. Pseudo-code for this algorithm is shown in Fig. 5.4, and successful assignment of objects can be seen on the right of Fig. 5.3.

5.5.2 Evaluation of Object Registration

We performed a simple evaluation of our object registration algorithm in order to determine the quality of the label positioning (registration). To do so, we laid out five objects on a table: a computer monitor, keyboard, scissors, plastic bottle, and a paper cup. We marked a target point on the desk from which to compare each object and measured the distance with millimeter accuracy between the target point and the center of each object using a tape measure. This measurement served as the ground truth (GT in Table 5.1) for our position estimation.

During the evaluation, a user stood in a fixed position in front of the desk wearing the HoloLens and was given a handheld input device (a small bluetooth keyboard). The user is asked not to move or rotate their body but only their head. The user is instructed to look around the desk until the mesh is constructed, which is indicated by the appearance of a blue cursor in the center of the display. They were then asked to look at each object and confirm that a label has been placed for each object. Afterwards, the user was directed to point the blue cursor onto the marked target point and click a button on the handheld input device. This triggers a raycast from the center of the display in order to determine the target point pose within the HoloLens'

Table 5.1: Data for ground truth (GT) and Estimation error in cm of the Euclidean distance between user-selected center points of each object in cm and a known 3D point in the tracking space.

Object	GT	User 1	User 2	User 3	Avg Error
TV	49.5	57.29	57.33	56.75	7.62
Keyboard	17.8	18.69	18.14	24.9	3.19
Scissors	50.8	50.3	51.06	51.32	0.90
Bottle	61	74.46	63.2	67.88	7.51
Cup	66	67.33	60.77	61.63	3.64
Overall					4.57

coordinate system. We then measure the distance between the estimated label positions and the target point and compare them to ground truth in Table 5.1. This evaluation was conducted by three users who had some prior experience with the HoloLens.

These preliminary results show that, on average, our object registration algorithm automatically converges on an object position up to 4.6cm away from the actual center position. Naturally, this is influenced by a number of factors, such as the size of the object to be labeled, and the initial vantage point when the label is first placed, but these values proved to be quite stable between users and repetitions.

In the future, we plan to evaluate performance on more challenging conditions. For instance, where the user is moving around the environment, or under poor lighting conditions. For now, the current registration performance is good enough for our needs.

5.6 Eye Tracking, Interaction, and Discussion

One more challenge in achieving a practical AR Language Learning system is the implementation of a method for selecting or activating an item for labelling. Simply labelling all objects in the environment is not feasible since the objects would clutter the user's view, so a method (either active or passive) for selection or specification is necessary. We believe the natural solution is an attention-aware interface such as eye tracking. Such an interface allows us to deliver learning content when the user is in an amicable state, and provides interaction without a cumbersome external device or difficult to use gestures.

In order to facilitate basic interaction with content, we implemented a calibration framework for our system to allow users to activate items via eye gaze. Though the evaluation of this area is a work in progress, we describe the implementation, how eye gaze fits into our overall framework, and several possible mechanisms for interaction below.

5.6.1 Eye Tracking and Calibration Module

Gaze based selection of objects provides an intuitive interface for managing AR content without the need for additional input devices or complex gestures. Since individuals almost always tend to gaze upon an item or object when learning through the method of loci, unknown concepts should be displayed quickly. In this way, our learning framework allows us to explore the effects of passive learning, in which educational content may be consumed throughout a user's daily routine.

Our calibration framework is based on the open source eye tracker built by Itoh et al. [168] for VR headsets, but with modifications made for the HoloLens. Much like a typical eye-to-video tracker calibration, we utilize a 5-point calibration interface in the Hololens. However, most eye tracking calibration procedures are executed with a sufficiently large field of view (FoV); i.e. the user gazes at several points on a 2D screen within the world-camera's wide FoV. In VR implementations, calibration points are often affixed to the display rather than registered in the world to counteract head movement. Since the Hololens FoV is only 35 degrees, we modified the same procedure used for VR and located vertical calibration points on the viewable portion of the screen. Though this can result in a minor reduction in vertical calibration accuracy, it sufficed for the purposes of activating labels on objects of interest.

5.6.2 Personalized Learning Model

The final component of our language learning framework is a personalized learning model. Specifically one that automatically adapts to the learners growth. We believe this is a fundamental difference between AR language learning and other existing language learning technologies. In our view, the future of augmented reality includes a collection of other vitals sensors which can monitor the physical and mental state of the user, similar to the trend of including health sensors in smartwatches. Already, we see devices like the Magic Leap One which include built-in eye trackers. This provides the ability to gauge the user's current understanding of the foreign language through continued monitoring of their cognitive response when consuming educational content.

As a first step, we plan to utilize eye and gaze signals, which have been shown to be good indicators of a user's point of focus. To validate a user's understanding of foreign words, we can use the duration of focus as an indicator of understanding. For example, labels that are gazed upon longer or multiple times within a short time period are likely to be unlearned. We plan to use these eye signals to develop a machine learning classifier that can detect whether a user understands or is confused about a foreign word. With such a classifier, we could identify how much foreign vocabulary a student has learned, and adapt by modifying the content (i.e, by introducing new words and removing words they have already learned).

We have recorded some preliminary results through a pilot study of 15 users. During the study, we presented English words in increasing difficulty to non-native English speakers while they wore a head mounted eye tracker. When presented with a word, the participants responded whether they did or did not know the meaning of the word. Afterwards, we developed an SVM classifier using the eye signals that was able to achieve 75% accuracy on the most difficult words. We plan to improve the performance by gathering more data and testing other classification techniques such as Recurrent Neural Networks.

5.6.3 Discussion and Future Work

Upon trying to implement a practical object labelling system in AR, we encountered many challenges that are not present in other object recognition implementations. For example, even though object recognition rates can exceed 90% on many 2D image datasets, this does not guarantee consistent use in the real world. Especially for a lower resolution camera that uses compressed images (such as the camera on the HoloLens), recognition from these algorithms is almost unusable unless modified as described in Section 5.5.1.

One other approach that we would like to explore is the re-training of object recognition models on video streams. Since integrated eye tracking in combination with the environment mesh can help determine the scale and depth of an object, we could potentially use this information to continuously re-train recognition for that particular object. User confirmation of recognition results also deserves consideration. For example, classification results may return the terms “tool” and “pen” for a ball-point pen. Allowing the user to select the term pen from a list could not only confirm the registered label in the immediate environment but improve recognition of that item upon the next encounter.

Our framework also tracks eye metrics such as pupil diameter and eye movement while users consume learning content in AR. As future work, we are investigating the use of machine learning based approaches to fuse and classify these signals for real time use. If we can automatically determine when a user understands a word, we can automate the learning algorithm used and suggest better, more relevant words to learn.

5.7 Conclusion

In this chapter, we introduced a framework for realizing in-situ augmented reality language learning. As part of this framework, we describe our current progress implementing a client-server architecture that provides the ability to conduct both object recognition and environment mapping in real time using a convolutional neural network. We explored the problem of object registration when using such a network, and provide a solution that accounts for the mismatched recognition errors that may occur. Our method is implemented directly on an AR headset. We described how to integrate eye tracking into our framework to allow for user selection or activation of annotations. We also described how to integrate a personalized learning model into our framework including initial results. We hope that this work will open up new avenues of research into methods and interactions for AR language learning and encourage others to contribute to this growing field.

We attempted to implement our language learning vision into a tangible prototype to the best of our ability, utilizing state-of-the-art techniques for eye tracking and object detection. Ultimately, the current state of research for these algorithms proved to be not mature enough to completely implement the system. However, the augmented reality community is in need of more in-depth analysis of machine learning techniques and how to incorporate them in mobile augmented reality headsets. In describing our ideas and efforts at implementation, we can posit the near-term feasibility of the system, and lay a path to follow for future application designers who wish to incorporate user context in their work.

Our framework shares some similarities but also notable differences to the Touring Machine prototype first presented in 1997 [41]. The Touring Machine demonstrated how to combine a collection of disparate AR systems and technologies into a novel prototype which enabled new AR use cases, namely mobile computing in large outdoor environments. In a similar manner, our work demonstrates how we can use AR for continuous and recurring interactions with the world, through a combination of hardware and software components that enable AR language learning. In that sense, our work can be seen as an extension of the Touring Machine work by considering the use of extended temporality (over the many interactive sessions) in the interface. Another notable similarity is the distribution of computation over several devices. In their case, the authors used a backpack PC to drive the display, GPS, and orientation tracker, while they used another handheld PC to drive the information database and access offline and online information sources. In the framework presented in this chapter, we also demonstrate how to distribute computaitons across a wearable display and a backpack computer. Except in our case, the display is now capable of driving itself and all of its required tracking, and instead we use the backpack computer to power the context-sensing and machine intelligence capabilities instead. It is telling that after 20 years of progress, we are still not able to reduce the form-factor substantially and continue to offload computational tasks. This is not a failure of hardware efficiency or software optimization, but rather, indicative of the vast amounts of information and data needed to power mobile AR systems. The Touring Machine was not concerned with the amount of information being used by the system, just about getting enough information at all to demonstrate a usable interface. Our framework today suggests that we may need to be more prudent with decisions on how much information is enough, so that we can begin to reduce the performance and computational requirements of mobile AR systems.

It is unlikely for AR headset to become a widespread consumer electronic on the level of smartphones and PCs, unless they can solve the context awareness problem. Without that, AR headsets will likely be resigned to special purpose use cases such as skilled labor or 3D modeling and visualization. Unfortunately, current AR systems still have a long way to go, and it is difficult to know where researchers should focus their efforts on. This problem is exacerbated by the increasing parameter size of deep learning models and the chase for superhuman performance on benchmark tasks. This work highlights the growing gap between the compute requirements of state-of-the-art context awareness models and AR devices. I hope it encourages researchers to focus more efforts on creating energy efficient systems and algorithms with low compute cost to better target AR headsets.

One meta contribution of this thesis is the practice of viewing current and future technical challenges through the perspective of long-term idealistic application goals, as I have done here with AR language learning. Though it is not a new practice, it is one that I believe the field has lost sight of in recent years. In my view, contemporary AR/VR and AI researchers are too focused on short-term results and small iterative changes. There is an emphasis on marginally beating the next benchmark, on developing research projects that can be spun off into start up companies and sold to the highest bidder. Instead, I urge researchers to think of what we want this technology to look like in the long term. 10, 15 years from now, what is our perfect vision of the augmented reality future. If we work backwards from that vision, we might have a chance of actually achieving it. But if we only focus on short-term goals, who knows what technological dystopia awaits us in the future.

Chapter 6

Situated Context Menus as a Model for Multitasking

We began this dissertation by identifying unique advantages of Augmented Reality compared to other mediums, establishing the value proposition or the ‘why’ augmented reality is important and how it might be beneficial for personal or domestic computing use cases. Next, we explored challenges to pervasive AR design using a top-down approach, focusing on an ideal use case and working backwards to identify long-term problems and contribute progress towards potential solutions. In the final stretch of this dissertation, we’ll take a more bottom-up approach, identifying core insufficiencies present in currently deployed AR paradigms and developing improvements that push the medium forward.

These chapters focus primarily on the problem of multitasking. How do we interact with multiple applications at a time? Existing AR headsets use an single-app paradigm that is more akin to smartphones than PCs. Smartphones typically only display one full-screen application at a time. More than one application may be executing operations on the CPU, GPU, or other hardware. But in practice, the user only experiences one application’s content, visual elements, UI elements, and interaction possibilities. Thus, a better term for this style of application might would be exclusive display. In contrast, desktops use concurrent display. That is, multiple applications can be displayed at the same time, and it is up to the user to switch their focus between each applications visual and interactive elements. This chapter specifically, presents an application model that uses object recognition to suggest applications that may be appropriate for the user given a specific situation or scenario. It takes inspiration from the context menu systems that exist in current desktop computers.

There is ultimately a more philosophical question at play as well. That is, to what degree should the user have control over the display of augmented reality content. Some proponents of pervasive AR believe that machine learning models should be widely deployed. That an always-on augmented reality device presents an infinite combination of interaction possibilities and situational use cases, to the extent that it is better off if an AI takes the lead and determines for itself what it deems useful for the user. This of course is not a new idea, but an extension of Nielsen's noncommand user interfaces [75]. Others might argue the opposite as well, opting for command-based interaction that places all of the control in the users hand. The answer, in my opinion, is somewhere in the middle. The real question we should be asking is, What is the right balance between automation and agency?

6.1 Introduction

As Augmented Reality gains popularity and moves closer to the hands of consumers, many questions still remain about how personal computing will actually occur on these devices. One of the most important features of modern personal computers is multitasking, the ability to switch between multiple running applications.

Currently available commercial HMDs, such as the Microsoft HoloLens 1, HoloLens 2, and the Magic Leap One, all have a centralized organizational system in which apps are launched one at a time through the use of a singular menu. While this may help lower the barrier to entry for new users, utilizing menus designed for traditional window-based systems comes with a variety of downsides. Likewise, application launch and management mechanisms used in modern smart phones or watches have not strayed too far from such manual switching between apps, one at a time, and have decided challenges due to the small screen space. This limitation is often referred to as the “single window constraint” and has been linked to lower multitasking performance on smartphones compared to traditional desktop counterparts, as the user is required to retain information from one application in their working memory as they switch to another [169].

In contrast, AR applications do not have to be limited to screen space and can potentially augment the entirety of a physical space with information. For example, the HoloLens allows users to place menus or other items using its airtap interaction, so menus can be physically tied to a location of choice. This type of annotation and menu system was demonstrated early on in the history of AR and refined over time [49, 52, 170, 171]. However, manually placing, inspecting, and manipulating menus in large open spaces may be even more mentally taxing and visually disruptive.

One of our goals in this paper is to improve upon existing task switching paradigms by designing a more adaptive in-situ menu system. In a future with always-on and contextually aware AR displays, we envision applications that depend on or react to the existence of physical objects and phenomena. In this case, switching between applications through a central menu could be awkward and cumbersome since users often do not know what applications might be available to interact with or augment real world items. For example, you might switch to a geocaching or wiki application, only to find that none of the objects in your immediate vicinity can interact with any installed applications. In order to test adaptive menus in context-aware settings, we had to develop a real-time object recognition framework which we also describe in this work.

Finally, concerns exist that AR applications may compete to monopolize a user’s attention.

Without an easy-to-use and democratic multitasking system, applications might be encouraged to augment the world indiscriminately, knowing that it is easier for the user to stay in the current application than to look for alternative ways to spend their time. In the same way that users spend an inordinate amount of time with social media on smartphones, the same may become a reality for AR applications that augment the physical environment. Applications may take a disproportionate amount of display space that could occlude and distract the user from real world tasks such as driving or locomotion.

With these issues in mind, we sought to design a multitasking system that would be more efficient, practical to use in always-on scenarios, and would make appropriate use of available screen and real world space. In short, our contributions include:

- a new multitasking interface that detects and reacts to the objects in the environment, demonstrated with the help of three context-based applications to test: a to-do list app, a workplace assistance app, and a vocabulary learning app.
- a comparison of this interface with existing approaches such as the aforementioned static menu app switching, which are tested with a novice user, domain expert, and system expert.
- a distillation of design considerations based on what we learned, along with a discussion of several new interface designs that build on our approach.

6.2 Related Work

Related work within the field can primarily be divided into three general areas. We start out on the more general side by discussing the field of context-aware AR interfaces. Related research in view- and content-management also helped motivate our approach. Finally, we review other systems that assist with or facilitate multitasking.

6.2.1 Context-Aware User Interfaces

Pervasive user interfaces describe computer interfaces that operate continuously and are aware and responsive to the user's context. Abowd et al. [172] provided a formal description of what constitutes context and context-awareness in user interfaces, which was further expanded by Schmidt [173]. Just prior to that, Starner et al. described the concept of mobile augmented reality [174], a precursor to pervasive AR, which included the idea of user modelling and adapting to user context. The Columbia Touring Machine [41] and its extension for situated outdoor storytelling [52] enabled the first outdoor AR information browsers including early application (story) switching mechanisms. The Studierstube AR system and application framework [175] demonstrated how AR applications can be distributed over, and collaboratively used from, multiple platforms, leading into the concept of social augmented reality [176].

More recently, Kim and colleagues conducted an excellent survey on the current state of AR research [1], highlighting pervasive interfaces as an emerging trend that is likely to define the next 10 years of research in the field. Around the same time, Grubert et al. [2] provided a taxonomy for pervasive user interfaces in AR, highlighting potential sources of context information and ways to act on that information to benefit the user. One example of the use of physical context is OmniTouch, a system that takes advantage of nearby objects to enable interactions [177]. Just like OmniTouch makes use of gesture recognition to enable projections on a user's hand, we similarly use object recognition to enable interactions with environmental objects.

6.2.2 Object Detection in Augmented Reality

Additionally, object detection will play a key role in context-aware AR interfaces moving forward. In order to interact with objects in-situ, the system has to know information about the object, and we have to assume that tagged (e.g. geospatially placed) information for an object is not likely to exist. There are several works [32, 178] that focus on segmenting dynamic objects from RGB-D point clouds in real time, often by extending existing real-time image-based detection algorithms such as Yolo [146] or Mask R-CNN [179]. Microsoft Azure and Google's Vision API also provide some functionality for doing object detection in images. However the adaptation of these detection algorithms for practical use with AR applications is still far from complete. Our work makes a step in this direction by effectively detecting and determining the correct location of objects for in-situ, real-world labeling in real time.

6.2.3 View Management

A number of works have tackled user interface management [180] in Augmented Reality. One such body of work focuses on view management, optimized label placement and appearance modulation. In a similar fashion to our use of 2D bounding boxes for recognizing objects in the image plane to determine a 3D label position for that object, several view management approaches optimize the placement of annotations based on the 2D rectangular extent of 3D objects in the image plane [152–154]. Other approaches allow the adjustment of labels in 3D space [155, 156, 181].

DiVerdi et al. proposed a system design for a marker-based AR window manager [49]. The authors consider several issues, such as input management and application focus, differences between physical and virtual applications, and inter-application communication. The authors extended this work [50] by considering different levels of rendering quality based on user dynamics.

Works have also looked at the layout of text and visual information in AR applications. For example, Grasset et al. [154] used visual saliency as a cue for driving the layout of text in an AR browser. We take a similar image-based context sensing approach, instead using object detection to drive view management. Moreover, our design of the icon size, placement, transparency, and leader lines draw from techniques developed in prior view management work. When a multitude of annotations is placed in front of the user, information filtering techniques are necessary to de-clutter the AR view [182, 183].

6.2.4 Multitasking in Augmented Reality

One key area in AR has always been the question of how to manage the user's time. He or she is always switching between a number of real-world and virtual tasks, and the ability to switch effectively, efficiently, and safely is paramount to solid interface design. For example, the Kimura system developed by MacIntyre et al. provided a way to view and interact with a desktop environment and wall sized display that share virtual windows for office tasks [184]. They present a number of different methods for arranging these windows that preserve the spatial arrangement and priority of data. Another system proposed by DiVerdi et al. described a design for a marker-based AR window manager [49]. The authors consider several issues, such as input management and application focus, differences between physical and virtual applications, and inter-application communication. The authors extended this work [50] by considering different levels of rendering quality based on user dynamics. Lages et al. [10] examined view management for multiple applications in an HMD walking scenario. They positioned windowed application views projected onto flat surfaces, and dynamically shifted the position of each application as the user walked. Lu et al. [11] looked at head-locked HMD interfaces that used eye-tracking to provide access to different application information by glancing in certain directions.

However, while they support multiple applications, these systems are constrained to 2D windowed application views. Multitasking was straightforward, as applications did not overlap in terms of their relevance to world-located objects and were small enough for the user to simply switch their focus by moving their head. In contrast, our work considers multitasking

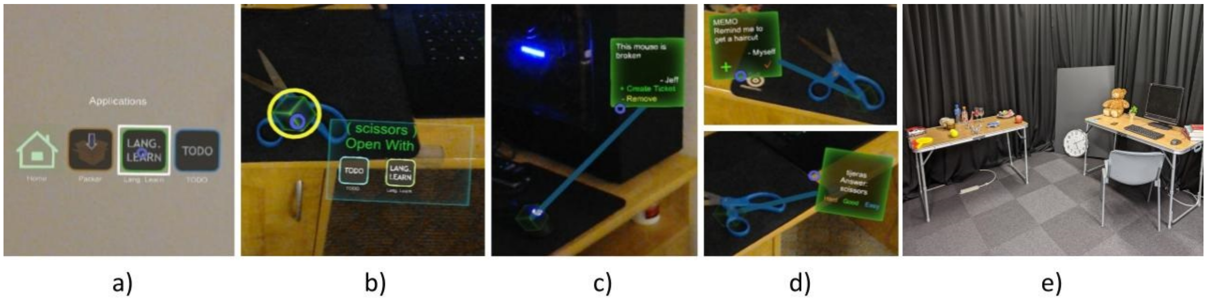


Figure 6.1: Initial prototype multitasking interfaces we designed, including a) Static App Switcher and b) In-situ App Switcher, along with three prototype AR applications including c) Packer: a workplace assistance application, d) ToDo: for managing day-to-day reminders (top), and Lang Learn: for foreign language flash cards (bottom), and e) the testing environment.

in environments where applications are capable of augmenting the entirety of a given space.

6.3 Methodology

This section describes our initial prototype multitasking interface. We implemented the interface using the real-time object detection framework for HoloLens described in Chapter 5, trained on on the MS COCO dataset to identify its 80 object categories. We developed three prototype applications that make use of objects within each category to provide augmented reality interactions. Examples of the interface design and testing environment can be seen in Figure 6.1, and details are further described below.

6.3.1 Hardware and Software Apparatus

We are using the Microsoft HoloLens as our AR head mounted display (HMD). The software running was developed in Unity utilizing Microsoft's Mixed Reality toolkit. The HoloLens was used in conjunction with an Xbox One bluetooth controller. The object detection procedure is based on a client-server modelthat communicates using UDP messages. Images from the HoloLens are sent to a remote server to be recognized, and then the relative positions of the recognized objects are returned to the HoloLens to be projected into world space. On the client side (the HoloLens), we first encode frames from the camera stream to JPEG before sending them to a server that was written in Python 3.6. The server receives client messages, decodes the camera frame, and hands it off to a separate thread for processing by a neural network. We used TensorFlow to implement this neural network, which is a variant of the Single Shot MultiBox Detector (SSD) [185]. The network is trained on Microsoft's Common Objects in Context (MS COCO) data set. We send the network output back to the client (HoloLens), where we project detected object centers from the 2D image onto the 3D mesh. On the client, we then filter and aggregate the neural network output in real time similar to the method used by our previous work on object recognition for AR language learning [72] in chapter 5, and provide that information in an easy to use event-based API.

6.3.2 Prototype Applications

We designed and implemented three different AR applications that operate on real-world objects, which can be seen in Figure 6.1.

ToDo: Interactive context-based Todo List

This application uses situated post-it notes to assist in the creation of a reminder list for everyday tasks, much like the work of Rekimoto et al. [186]. In this "ToDo" app, the user can see virtual, in-situ messages from friends and family members that have been placed throughout the environment on various objects, just like a "Post-it" or sticky note. These notes are coupled with physical objects in the environment to provide additional context for a potential task the user might want to add to their todo-list. For instance, a note from your wife reminding you to pick up groceries after work would be affixed to specific grocery items, such as fruit or wine bottles. The user can choose whether to add these task items to their todo-list or dismiss them from the AR application.

Lang Learn: In-situ Language Learning

Augmented Reality is a prime candidate for assisting with vocabulary or language learning. The learning of new words is often conducted in-situ, which is usually referred to as the “method of loci.” This means that learners are better able to absorb the meaning of a word if that word is learned in context, i.e., when the object is physically present at the time of learning. A number of works in AR have looked into this effect, including [70] and [187].

We implemented a new application that can take advantage of this phenomenon. In the “Lang Learn” app, physical objects are labeled with a flash card showing the word’s name in a target foreign language, in this case Spanish. The user is tasked with remembering the English translation. Once they feel they have remembered the correct translation, or if they give up, they can click a button to show the answer. Afterwards, they are asked to rate the degree of familiarity with the answer, a technique commonly used in spaced-repetition-based learning applications [188].

Packer: Workplace Assistance

The third application is a typical workflow assistance application that might be used in a workplace such as an office or shipping company. “Packer” assumes that the user is an employee in a packing and distribution warehouse. In such an environment, AR could be helpful for identifying or locating a specific object within a large and cluttered space. The premise of our app is that the user is a facility manager in such a warehouse. They are tasked with locating defective objects indicated by other workers, and confirming whether they are indeed defective or not. The application labels these objects with relevant comments from workers, and the user can choose to create a work ticket to track resolution of the problem.

6.3.3 Multitasking Interfaces

Our *in-situ* app switcher system attaches menu options directly to an object via co-located menu icons, as shown in (b) of Figure 6.1. To compare this with a more traditional approach, we implemented a *static* app switcher, which mimics menu systems found in existing HMDs, shown in (a) of Figure 6.1. The implementation details of each are described below. Additionally, in experiments, we test a hybrid combination of static and in-situ app switching where the user can freely choose between each as convenient for the situation.

Static App Switcher

The static app switcher is a variant of the traditional screen-centered menu systems in the HoloLens and Magic Leap interfaces. The user can press the A button (on the controller provided in our experiments) to bring up or send away this menu. When summoned, the menu appears directly in front of the user's field of view and is fixed in place in 3D, and remains in that location until the user sends the menu away. To open a menu item, the user can gaze at the item within the menu and press the A button. The menu is sent away automatically once an app is selected.

In-Situ App Switcher

Our in-situ app switcher is a novel extension of context menus for traditional 2D GUIs, by presenting them in association with physical objects and situated in 3D space. Our in-situ app switcher provides a list of all applications that have content relevant to a particular object and is unique to each object. To activate, the user gazes at the object marker and presses the A button on the controller to bring up the in-situ menu. Moreover when the user's gaze intersects with the cube, a green circle will appear and the brightness of the cube will be increased (transparency reduced), signifying that an application has relevant content for that object.

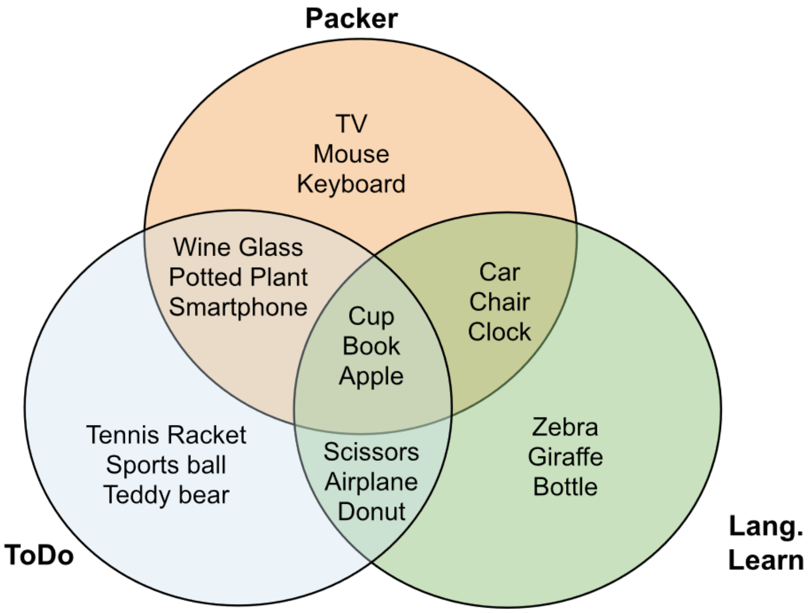


Figure 6.2: Venn diagram of interactive objects and which applications provided content for those objects. This design ensured that groups of objects have content relevant to 1, 2, and 3 applications.

Once the user brings up any relevant in-situ menus, he or she can gaze at the menu item and press the A button to select an application within that menu. Our context menu design utilizes the same rectangular panels as the static app switcher menu for objective comparison, but they always appear directly to the right of the object marker. Additionally, all of the panes are presented in billboard style to ensure readability.

6.3.4 Pilot Study: Novice vs. Experts Comparison

This study was designed to evaluate the performance of different multitasking presentation strategies for switching between applications. To do so, we set up a test where the user had to conduct language learning, update a to-do list, and carry out work tasks (inspect and submit tickets) within our three representative AR applications. These applications were applied to the test environment shown in (e) of Figure 6.1, and used the aforementioned object detection algorithm on the objects in the environment. We assigned 12 object subtasks to each of the three representative AR applications. Some were associated with multiple applications, while some only worked with one specific application. This was distributed as equally as possible as shown in the Venn diagram in Figure 6.2, leading to a total of 21 objects with 36 subtasks. The user's goal was to carry out these subtasks as quickly as possible with each of the multitasking interfaces.

In the experiments, we were interested in how our three presentation methods, as described in Section 3.3, would affect multitasking performance and usage in the test environment. To reiterate, the methods were:

- *Static App Switcher* - App-drawer approach where users select another application by first opening a static menu that appears directly in front of the users head.
- *In-situ App Switcher* - Situated approach where users select from a list of compatible applications for an object through a menu which is placed directly next to each object.
- *Combined Selection* - In the combined approach, the user can activate either type of menu and use them interchangeably.

The goal of the study was to get through the all of the subtasks as quickly as possible with the given interface. The same novice user, domain expert, and system expert that participated in Experiment 2 also participated in this study. They completed each of the 36 subtasks for each of the three multitasking interfaces, starting with the Static App Switcher, followed by In-situ App Switcher, and finally Combined. Participants were instructed to complete the subtasks as quickly as possible while still reading the text in each of the apps from start to finish. Note that participants were given a training period of 10 minutes before the study to get accustomed to the HMD and learn the controls for each interface.

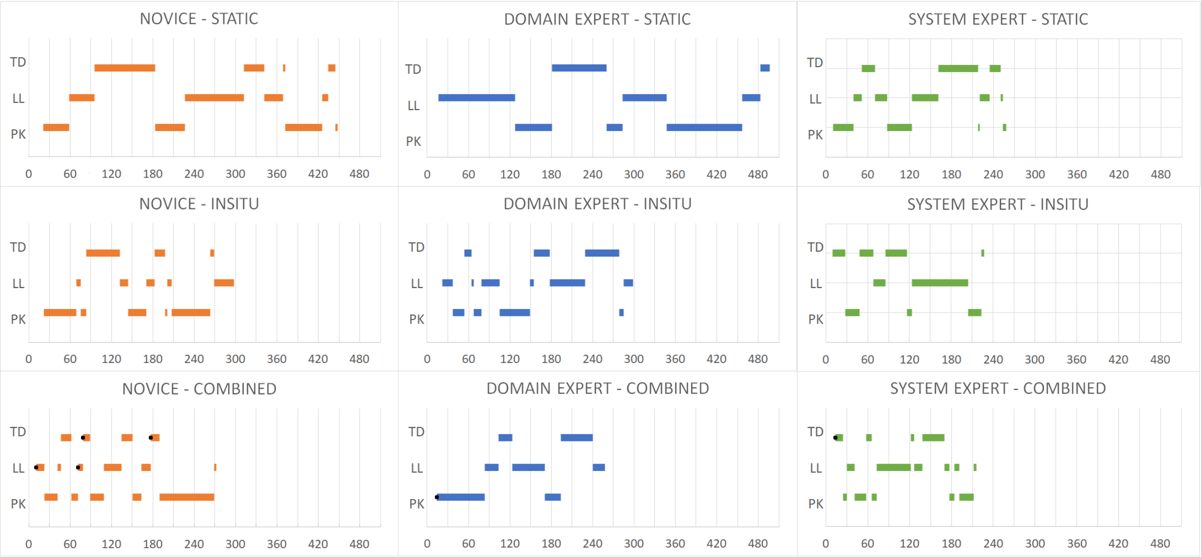


Figure 6.3: Timeline of usage for applications ToDo (TD), Lang. Learn (LL) and Packer (PK), in seconds. Columns from left to right are Novice, Domain Expert, and System Expert participants, while rows from top to bottom are Static App Switcher, In-situ App Switcher, and Combined interfaces. For combined interfaces, a black dot at the start of the segment indicates using the Static App Switcher, otherwise it was In-situ App Switcher.

6.3.5 Quantitative Results

We first calculated how much time each participant spent with each application as well as how many switches occurred. A detailed breakdown of application usage is shown in Figure 6.3 as a 3x3 matrix of timelines, allowing us to see switch frequency and observe usage patterns. Participants averaged 11.4 switches in order to complete all 36 subtasks. The domain expert averaged 9.3 switches, the system expert averaged 12, and the novice user averaged 13.

In terms of completion time, we observed that both the novice and domain expert took longer using the static app switcher compared to other interfaces. The system expert also took longer using the static app switcher, though the difference was not as pronounced. It is difficult to determine if this could be due to order effects. However, the fact that the system expert also took longer despite having extensive experience supports the observation that at least some of the difference in duration of the static interface is not due to ordering effects.

By looking at the rows in the timeline matrix, we can compare usage patterns between each

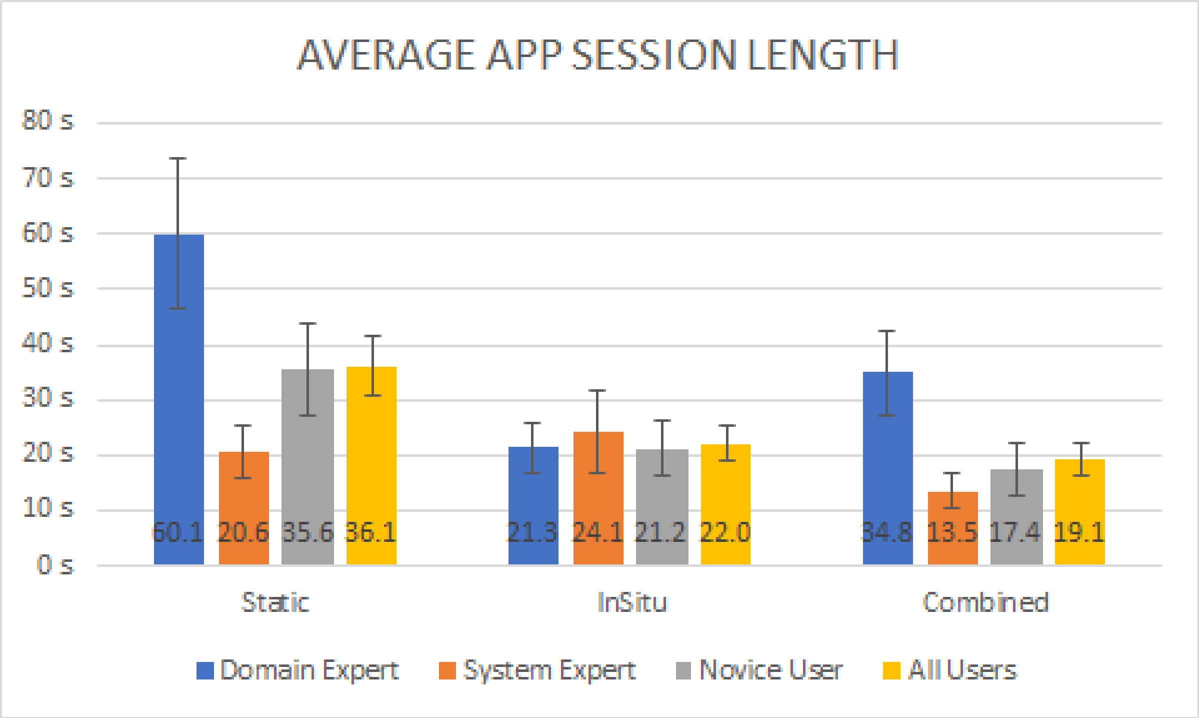


Figure 6.4: Average app session length w/ standard error . Yellow is the average over all users.

condition. We noticed that participants tend to stay in apps longer with Static App Switcher. In the other two conditions, participants tend to use apps for shorter periods of time. This can also be seen in Figure 6.4, which summarizes the average app session lengths. Looking at the average session length for all users, we can see that it was higher when using the Static App Switcher versus the In-situ App Switcher or Combined conditions.

The bottom row of Figure 6.3 shows usage during the Combined condition, in which participants were free to use either Static App Switcher or In-situ App Switcher at any time to switch apps. Both experts only used Static App Switcher once, at the very start of the session. The Novice user also used it to start the session and used it three additional times. However, 73% of their app switches were still through the In-situ App Switcher. Despite having both options available, participants preferred to use the In-situ App Switcher a majority of the time.

6.3.6 Qualitative Results

At the end of Study 3, we conducted a semi-structured interview with the novice and domain expert. The goal of this interview was to determine which interface the participants preferred and to see if there were any notable issues they struggled with. Topics for the interview were grouped into the five categories below.

Contextual Consistency. When asked *Was it clear at all times which app you were in? Why or why not?*, our domain expert responded that it was "kind of hard to understand what kind of app I am in, probably because all the UI are the same color". The novice user mentioned having to frequently check the icon in the corner to know which app they were in. They also mentioned that "the content is kind of the same based on the app", indicating that content differences could be used to determine which app they were in. In general, participants did not seem to think any of the multitasking interfaces had a large effect on their understanding of the current app context.

Attention Management. When asked *Did you prefer to spend a lot of time in one app or did you prefer to switch between apps often?*. The domain expert felt that "For the todo and the language learning I prefer to stay quite long because I want to complete it as much as possible." He also noted while using other applications that he would sometimes see an object and get really curious about the Spanish word, and choose to switch to the language learning app right away, mentioning "The [context] menu at the box is easier for this." The novice user stated that their strategy was to "do everything I could see in one app. Then I would use the little diamond [cube] menu on the closest object to see if there's anything left."

Status Visibility. We asked *Was it easy to understand which objects had augmentations remaining? Was one better than the other?*. The domain expert responded that he had an easier time understanding how many subtasks were left with the In-Situ App Switcher and Combined interfaces. The novice user also felt that only with the in-situ menu could she "tell if there was anything left." When asked *Did it ever feel like you were missing some augmentations that you couldn't find?*, the domain expert responded "yes a little bit, because of the object recognition." While the novice user said they had difficulty finding augmentations with the Static App Switcher, stating "yeah with the first one I was going from right to left... there [were] two left [but] I didn't know where."

Natural Interaction. When asked *Which modality felt more natural to you?*, the domain expert responded with In-situ App Switcher. The novice user response was also the same, mentioning it felt natural to fidget with it when they weren't sure what else to do. When asked *Did you feel like you did a lot of unnecessary switching in any modality?* The expert responded with "Not really.", while the novice user responded "maybe in the first one because I was going through the apps in order", referencing the above comment about it being difficult to find the last two objects. This behavior can be seen in the timeline in Figure 6.3 where the last three app sessions in the Novice-Static graph might be short bouts of searching in each app, distinct from the rest of the session. When asked *Did one feel faster than the others*, the domain expert responded with the In-situ App Switcher and Combined options, while the novice user responded with the Combined option. When asked *Did you feel more productive using one vs. the other?*, the expert responded with the In-situ and Combined options while the novice user responded with the Combined option.

Overall Preference. When asked *Which interface did you prefer the most and why?* our domain expert responded that they preferred the Combined option as it was the most flexible. He also felt that the In-situ option would work just fine, as he felt he used the context menu the majority of the time. When asked why he preferred to use the context menu most of the time, he responded "It's good because when I see the object I was kind of remind [sic] what I have left [to do]. In terms of function I'm gonna use the menu on the cube a lot." The novice user responded that they also preferred the Combined option as well for similar reasons. They mentioned using in-situ menu most of the time "to check what was left if I couldn't find anything else to do." This is backed up by our quantitative results, which showed that both participants used the Static App Switcher very infrequently during the combined session.

6.4 Design Considerations

Based on comments and findings from our user study, as well as our own practical experience in developing and using these interfaces, we identified some important points to consider when designing multitasking systems for AR. These design considerations can be used to help design multitasking systems for AR content in the future, especially for devices that may have a number of AR applications running at once.

6.4.1 Contextual Consistency

Because of the immersive nature of 3D Situated applications, it can be difficult to actually know which application you are in when frequently switching between multiple applications with different content. This feedback was common among our study participants. One participant suggested having different colors to represent different applications. However, when implementing these applications, we realized that requiring a unique design for every app is not always feasible or desirable. Due to the limited FOV of current generation AR HMDs, adding too many colorful visual design elements could lead to clutter and distract from the real world. One element we did add was a HUD icon in the top left corner that identified the current app, but this did not seem to help our participants considerably. Instead, one way of maintaining contextual consistency could be through predictable content. One participant mentioned that the thematic similarity of the content within an app helped her to recognize which app she was currently using. Text labels on the app content itself may also be useful for this purpose.

6.4.2 Context-dependent Functionality

Due to the cognitive costs of task-switching, friction-less multitasking will likely be invaluable for context-aware AR applications. These applications are likely to rely on the presence of specific objects in the environment, and it would be frustrating to open apps and find out they don't work properly because those objects are not considered. There could also be scenarios where, while using one app, changes in the environment trigger new content in another app unbeknownst to the user. An effective multitasking system should allow the user to quickly check for available content in all apps. In the post-study interview, our domain expert also suggested inclusion of an accompanying notification element to summarize or provide hints about the content.

When designing our In-situ App Switcher, we considered two different approaches for presenting context-dependent app content:

- *Object content only* - When selecting the app in the in-situ menu, only content related to the detected object is displayed. If the user moves away from the object, the app is suspended and returns to the previous app.

- *All content* - When switching apps, all available content for all currently detected objects is displayed. Moving away does not suspend the app.

We initially thought the object-content-only method would allow users to preview the app content for each object and easily jump in and out for short interactions. We tested both in a small pilot study but ultimately found that with this method, users frequently lost track of which app they were in and which objects they had already interacted with. Based on these experiences, we went with the all-content approach.

During the course of our research, we identified a few objects that were difficult to detect for users new to our system. The challenges can be broken down into two categories of problems: scale invariance and context of use. First, many state-of-the-art object detection networks are not fully scale invariant, including the one we used. The network struggled to detect objects that are typically seen at large scales, for example cars and giraffes. Second, our network was trained on the popular MS COCO dataset which emphasizes objects in their context of use. This means objects such as donuts or apples are frequently shown surrounded by other foods, which was not the case in our test environment. These remain important challenges for improving the stability and reliability of object detection algorithms for use in real-time AR applications.

6.4.3 Natural Interaction and User Well-being

In considering always-on AR devices, it is important to design solutions that feel natural to interact with. One comment we heard from participants was how easy it was to open and check apps with the In-situ App Switcher and how they were compelled to play with with nearby in-situ menus during idle moments. There may be something natural or organic about contextualizing the point of entry for an application closer to the object itself. Additionally, as we were implementing more and more object intelligence into our system, we were naturally compelled to try picking up and moving around objects in the space. Picking up objects can explicitly signal a user's intended activity and acts as a natural filtering mechanism for apps. This level of context awareness is difficult to achieve with current technologies but could prove a useful vector for multitasking strategies in the future.

3D Situated applications can run the risk of monopolizing too much of a user's attention, potentially distracting them from real world tasks. Additionally, multitasking in general has been shown to have a high task switching cost [189], which is the time and cognitive effort it takes for humans to adjust their mental control settings from one task to another. This effect could be even more pronounced in AR. As we move forward in the design of these systems, we need to ensure that we prioritize user well-being.

6.5 Discussion and Future Direction

In this chapter, we focus on the design and evaluation of a multitasking technique for pervasive augmented reality. The work addresses an important challenges when using augmented reality everyday in a variety of different situations, namely how to effectively switch between different applications, moving away from single-purpose AR. This solution was inspired by right-click context menus which first appeared in desktop graphical user interfaces in the 1970s, and are ubiquitous today. Our version applies this technique to individual objects, anticipating a shift towards context-aware applications that utilize said objects in their functionality.

We identified several promising directions for future work. One potential multitasking strategy is the creation of a common standard for presentation and placement of augmented reality content. This would look similar to layout engines in modern web browsers, allowing AR displays to arrange the content in a user's space by optimizing a set of display constraints among the user, device, available objects in the environment, and the applications themselves. Another option would be organizing applications into layers over the physical space, much like layers in an image editing program. Each layer could link one or more applications into meaningful groups. Applications can augment the space at their own discretion, but only the currently visible layers would be seen. The user would use a quick-access menu to toggle on and off visible layers at any given time. Yet another option is to segment different portions of the physical space itself and restrict each app to only display within those geometric bounds. In this model, apps first start by requesting the user to "slice," or indicate an area within the current environment. This could be done easily by moving and placing a cube or view frustum. This also provides an opportunity for the user to preview the content as they are deciding where to place the app. Once apps start, they are only capable of displaying their content within that predetermined space.

This work only explored object-level context awareness, but other sources of context could be used to improve multitasking. For instance, the HoloLens 2 provides eye-tracking and hand-tracking sensors. Eye-tracking can be used as a context cue to indicate user's attentive state [190] or determine salient points of interest [191] in the environment. Hand motions offer contextual information about a user's current activities and state of motion [192]. Other sensors could be integrated, like GPS or heart rate, to contextualize general location and physical state. Our image-based approach could be extended to incorporate detection of events [193], scene characteristics [194], and even social constructs [195].

As researchers working on the cutting edge of new technologies, we have a tendency to believe technology is a panacea that can cure all problems. Newer is always better, or so the thinking goes. Perhaps it is this mindset that has pushed many fields of computer science towards machine learning as the next big solution to all our problems. After all, it is newer, and it has outperformed all previous solutions thus far. But just because it is newer, doesn't mean it is better. Newer methods often come with unintended consequences. Machine learning relies on massive amounts of data collected from humans or annotated by humans, which can exhibit biases especially towards the socioeconomically disadvantaged. The training of ML algorithms take massive amounts of time, computing resources, energy, and money, with significant impact to the environment. The price to create state-of-the-art machine learning models is so cost-prohibitive that only the most wealthy companies can afford to invest in them.

Given these inequalities and other potential downsides to large machine learning models, it is crucial that we step back and think about how we deploy large ML models and what role they play in our AR future. This work proposes some potential benefits of using ML models as a form of information filtering and recommendation, while still allowing user agency through the actual choice of which application to launch. Is that an appropriate boundary? Or would another work better? We don't have enough information to know at this point.

Chapter 7

Layerable Applications as a Model for Multitasking

In this last chapter, we look at the design of a multitasking system and application model called Layerable Apps. The goal of this work is to provide alternative app models to those that already exist on current AR systems, as well as those that are being proposed by large tech companies for the future, such as "Metaverse" concepts [5]. The current trend of app models seems to favor large one-stop shop style applications where a single application will compete for users attention. This makes it exceeding difficult for smaller developers to make an impact in the AR space, and increase the likelihood that AR use cases of the future will be dictated by entrenched companies with large amounts of cash reserves needed to create as many features as possible into a single application. Additionally, that means these individual companies are likely to be the single point of failure for all your private and personal information that is collected in their application.

Instead, we developed the concept of Layerable Applications, an application model that supports multitasking and interaction with multiple applications of smaller scope. Our system supports different styles of application models, providing a starting point for further development of interoperability methods between disparate types of applications. Additionally, we also look at different levels of augmentation and increased user control of augmentation amount, and find some evidence to suggest different users have different amounts of augmentation they can tolerate. Ultimately, these findings rebuke the notion of a one-size-fits-all AR application.

7.1 Introduction

Today's AR systems are often operated in a single-application paradigm, in which users switch between one active application at a time. Though this model is good for interacting with individual pieces of content, it is not suitable for interaction with and viewing of multiple applications that might be displaying content using different modalities and might need to be cross-referenced with each other. For example, one application may require the use of a pin-pad for text entry, whereas another may augment existing waypoints with annotations. Current devices require the user to switch from one application to another, despite the fact that the applications may be used together, such as note taking during navigation.

In this work, we propose and evaluate the concept of Layerable apps: applications which can be quickly and easily layered on top of each other by the user. Layerable apps provide an increased degree of control and granularity, allowing the user to decide how much of their world is augmented, while still being able to perform tasks that integrate information between multiple applications or require simultaneous interaction between them. One of the goals of this paradigm is to provide a more consistent user experience in which interaction is seamless and application switching is less noticeable. Our primary research questions include:

- Do Layerable Apps provide advantages for multitasking performance?
- What effect does the use of Layerable Apps have on users' application usage and spatial awareness?

- What do user preferences look like when presented with Layerable Apps vs. traditional approaches?

In service of these questions, we implemented a prototype system consisting of an application switcher and four example applications, which are shown in Figure 7.1. This system allows for application switching via exclusive display (i.e., the currently adopted application switching scheme in most AR operating systems) and concurrent display (our Layerable Apps approach). We designed an experimental task with 44 participants that required users to actively engage with each application, and we used quantitative and qualitative methods to examine how users interact with multiple AR applications under the Layerable Application model.

7.2 Related Work

Related work primarily falls into two categories, including research that seeks to develop Augmented Reality as a personal computing paradigm, and view management systems that deal with menu placement and interaction.

7.2.1 Augmented Reality for Personal Computing

Throughout the development of AR technologies, one goal has been to integrate AR systems into everyday life as a type of personal computing device. For example, Starner et al.'s conceptualization of an augmented reality wearable interface [196] focused on the use of wearable AR as an assistive technology, acting as a kind of extended memory for the user, capable of storing and retrieving timely information.

A recent survey by Merino et al. [197] provided a comprehensive review of Mixed and Augmented Reality research and identified pervasive and always-on AR as a growing and important topic. Grubert et al. laid a foundation and taxonomy for describing this type of work, termed pervasive augmented reality [2]. Many works have examined individual application scenarios targeting everyday consumers, for example interior design [198], cooking [199,200], and retail shopping [82,201].

One such application by Knierim et al. utilized a technology probe to explore the potential of augmented reality usage in the home [202]. They found that most domestic participants were very accepting of AR as a personal technology to be used in domestic spaces, although they had some concerns about privacy and transparency. They identified potential use cases, including the use of AR to support everyday activities like grocery shopping, and the enhancement of everyday objects with new AR functionality.

Our work builds on these usage scenarios by investigating user behaviors and expectations for how to switch between these applications, and how to design the interfaces such that they can operate in seemingly seamless and non-obtrusive ways.

7.2.2 Information Placement and AR App Management

More recently, researchers have begun to more thoroughly explore different interfaces and paradigms for interacting with multiple information sources, including the simultaneous integration of menus, annotations, and augmentations in the same environment. One of the early attempts at managing a user's view was the work by Bell et al. [152], which allowed for improved placement of text and images such that all content was viewable. Hoang et al. developed a similar system for interacting with in-situ 3D objects from world-relative and head-relative in-situ menus [203]. Probably one of the most comprehensive menu systems was that of Brudy et al, who came up with a number of different menu styles that allowed for in-situ selection and manipulation of menu items [204]. Though not a menu system, Ubii provides for interaction with and selection of icons or other widgets in-situ [205]. Pourmemar took this a step further and developed hierarchical menus that could be used to select from multi-level lists as well as conduct manipulations [206].

In addition to menu-based interaction, context sensitivity has often been integrated into information presentation in AR. Integration of context or context awareness is present in many applications, such as location detection for relevant content placement [207], activity detection for AR video instruction [208], face detection for conversation-based AR [209], and object detection for in-situ language learning [72]. Other interfaces such as Glanceable AR allow for a combination of context and natural glance-based interaction for easy information access [11].

On the commercial front, both the Microsoft HoloLens 2 and the Magic Leap AR headsets have implemented limited forms of multi-app management. On the HoloLens 2 users are limited to certain combinations of a single 'mixed reality app' and a single '2D view' app alongside it [210]. Magic Leap has a 'Landscape' experience which allows multiple apps to display simple 2D content only [211].

While these systems provide a variety of different ways to interact with and view individual applications or specific groups of applications, the management of multiple applications that may be constantly available to the user is still not well explored. Lebeck et al. identified the problem space of multi-app AR laying the foundation for our work [212]. They suggested user-managed application output as a potential solution to the challenges of multi-app AR, which is our central focus.

Our work seeks to address this problem by determining what methods of application activation are most effective for dealing with multiple AR paradigms that are simultaneously available to the user. Simply put, we ask if it is better to manage applications through currently available menu systems that launch apps that take over the user environment exclusively, or if an in-situ layered approach may be more effective?

7.3 Layerable Applications

Augmented reality can often be described as the layering of digital information on top of the real world. Many futurists envision this digital layer to be a monolithic application that services all the needs of a user. For example, the concept of the Metaverse, where an AR user would engage with a single shared digital layer for all their entertainment and productivity needs, as is the current vision of companies such as Meta and Epic Games. While such efforts are necessary, they are also susceptible to privacy and security implications and could significantly limit users' technology choices, while giving an unprecedented degree of personal access to the companies and stakeholders who own the Metaverse platforms.

In this work, our goal was to explore AR applications not as monolithic do-everything systems, but as smaller, single-purpose, modular elements, with the goal of empowering the user to decide to what extent they want to engage with an augmented world. For this purpose, we came up with Layerable applications, which treat content as a series of "layers" on top of the physical world that can be toggled quickly and seamlessly. Multiple application layers can be used at the same time. This encourages the creation of applications that are still singular in scope, but allow the user to mix and match preferred functions depending on the situation.

When approaching the design and evaluation of Layerable applications, our goals were to (1) create a working prototype system capable of simulating the experience of using Layerable apps, (2) create a set of example applications to implement within the prototype system, and (3) develop an experimental task that required users to engage with each application modality to solve tasks.

7.3.1 System Design

We implemented a prototype of Layerable applications using Unity and deployed it to the Microsoft HoloLens 2. The system features an application menu that is brought up by looking

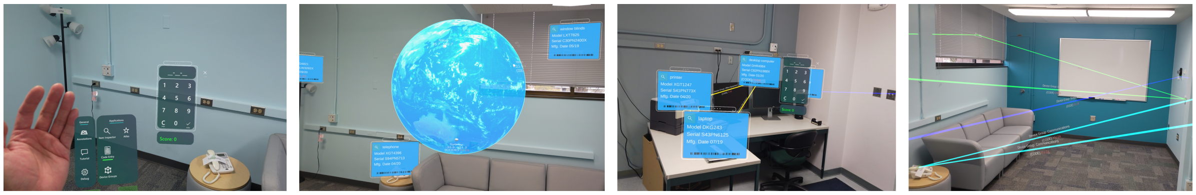


Figure 7.1: Images of the Layerable Apps paradigm showing (left to right) Code Entry and the Application Switcher, Atlas, Item Inspector, and Device Groups, all of which can be displayed concurrently to facilitate passing of information among them. In experiments, users were required to switch between and integrate information from all apps using different methods of application selection.

at the palm of your hands. With one hand, users can bring up the application menu, and with their other hand, they can tap the application icons to toggle the respective application layer on and off. Currently open layers are indicated with a green underline, as shown in leftmost image in Figure 7.1. The menu is ambidextrous and can be viewed on either hand.

When the system is in *Layerable* mode, application layers can be toggled on and off based on user preference. Users may prefer to use more or fewer applications, or to activate certain applications which have higher or lower amounts of augmentation, depending on their goals and physical situation.

Our system also features an implementation of the single-focus app model for the purposes of comparison in our user study. This model, which we call *Immersive* mode, imitates the behavior of applications in most contemporary AR headsets. In this mode, apps are launched one at a time, and opening an app will suspend any other currently open app. We chose to re-implement this behavior within our system instead of using HoloLens' default application launcher to provide a fairer comparison, as a) the HoloLens performs other operating system tasks that dramatically increase the time it takes to open an application, and b) this choice allowed us to use matching visual identities for the UI design of either mode.

Table 7.1: Application Categories

	Context-aware	Context-free
2D Presentation	Item Inspector	Code Entry
3D Presentation	Device Groups	Atlas

7.3.2 Representative Applications

The visual presentation and interaction capabilities of an AR application can vary significantly depending on the intent of the application designer. There is no widely accepted standard by which AR apps should look and feel. This makes it difficult to implement meaningful exemplar applications for testing an application switching system. With the Layerable Apps prototype, we identified a minimal taxonomy for the most common styles of AR applications that the system should support. We identified 2D and 3D as modalities of application presentation. 2D applications are those where all of their graphics are rendered within the confines of a 2D plane, though the plane itself may exist in 3D space. Notably, this encompasses all instances of traditional applications found on desktops and touch-screen devices, making it plausible to port those applications into our Layerable App system (a pathway that Microsoft has outlined for their 2D windows universal platform (UWP) apps and Windows Mixed Reality). 3D applications are all other applications that render graphics at multiple 3D positions. We found these categories to be representative of nearly all types of AR applications found on current commercially available HMDs.

Additionally, we wanted to incorporate some element of context awareness into our design. This was inspired by Grubert et. al's work on pervasive augmented reality [2], which suggests that future AR applications are likely to feature context-sensitive functionality. We chose to further categorize applications by whether or not they utilize context. Thus, our final design includes four example applications, with each app representing one of the four possible combinations of context-awareness and spatial interaction (cf. Table 7.1).

After enumerating the desired application types to support in our system, we implemented representative applications for each category. When conceptualizing the design of these apps, we tried to think of functionality that would be desirable to users in a real world setting. Images showing the contents of each application are shown in Figure 7.1. The applications we arrived at over many iterations and pilot evaluations are the following:

Item Inspector. Inspired by Internet of Things (IoT) applications, Item Inspector allows you to visually inspect the status and associated technical information about objects and devices in your home, such as battery life, model number, and manufacturing date. Information about each object is displayed within a 2D plane fixated above the object itself as shown in the 2nd and 3rd images in Figure 7.1. Object locations were tracked using manually placed spatial anchors in our controlled testing environment.

Device Groups. Using device groups, participants can group physical objects together in their space in order to perform aggregate actions such as turning all devices in a group on or off. Device groups are visually represented with colored lines connecting every object in a particular group to every other object in that group. These lines are rendered in 3D space, allowing the user to quickly grasp which objects are part of a group and where their locations are in the space.

Code Entry. This app enables users to virtually enter passcodes and pin numbers in place of traditional keypads on door locks, ATMs, and other security systems. In practice, the app functions similar to a calculator, displaying a number pad on a 2D plane, but not with any spatial dependency on any specific objects in the environment. This makes it suitable as a representative for a context-free and 2D application.

Atlas. Atlas displays a large 3D model of planet Earth that users can explore, displaying geographical information about cities and landmarks around the world. The model is rendered intentionally large – it can be scaled within certain limits but maintains a minimum size so as to 'fill' the space and require users to walk around when looking for a particular location.

7.3.3 Experimental Task

Our goal was to design an experimental task that would require the user to engage with all four applications in order to complete the task. We chose to employ a split-information task where the necessary pieces of information needed to complete the task are split up and distributed to each representative application.

In the study, users were tasked with finding pieces of a six digit code. Each code was split into three code fragments and each fragment was embedded into random 'flavor text' within the Item Inspector, Device Groups, and Atlas applications. Each code fragment also featured two leading alphabetical characters to help identify which fragments belonged to the same code. For instance, the user might encounter the fragment SC-12 in one app, SC-23 in another app, and SC-89 in a third app. After finding all three code fragments for a corresponding code, users could enter the digit pairs into the Code Entry application in any order. Participants were scored by the number of correct codes entered within a fixed amount of time. There were no penalties for incorrect codes (apart from the elapsed time used to enter them).

7.4 Experiment Design

We conducted a within-subjects user study with 44 participants over the course of two weeks. The study was conducted primarily with students and affiliates at a university campus and included students from different departments as well as local community participants signed up with a human subjects pool managed by the university. Study sessions took approximately 1.5 hours to complete. The average age of participants was 22.9, with 16 male, 25 female, and three identifying as non-binary.

7.4.1 Procedure

Participants filled out a demographics questionnaire and consent form prior to arriving for the experiment. Upon arriving, participants were trained on how to perform hand gestures within the HoloLens. Specifically, they were taught how to press a button and how to air tap on buttons that were far away. Participants used a training application that provided multiple opportunities to test their ability to execute the gestures correctly. Participants were asked if they were confident in their ability to execute the gesture before proceeding.

Following training, participants were placed into the Layerable Apps prototype and provided with a guided tutorial on how to complete the experimental task. The tutorial provided step-by-step instructions with text bubbles and text-to-speech voiceover, demonstrating how to open application layers, how to find codes hidden in each application, what the structure of the codes were, and how to input them. Participants were required to find and enter a code successfully to complete the tutorial. Afterwards, participants were asked to verbally describe to the experimenter, in their own words, what the task was and how to complete it.

After completing the training and tutorial, participants performed four task sessions of seven minutes each, alternating between *layerable* and *immersive* modes. Participants were counter-balanced with respect to their starting application mode. After the 1st task session, participants were asked to fill out a post-task questionnaire to capture their thoughts on the usability of that mode, as well as an object recognition quiz and object placement quiz where they were asked to recall information about objects in the scene. After the 2nd task session, participants were administered another post-task questionnaire to capture their thoughts on the alternative app switching mode. Finally, a post-study questionnaire and semi-structured interview was administered following the 4th and final task session. Throughout the study, modes were coded to “Mode A” for layerable and “Mode B” for immersive to avoid name bias.

To validate the sufficiency of our training procedures, we asked participants to rate their understanding of the tutorial, the task, and ease of use of the HoloLens, on a 7-point Likert scale.

7.4.2 Metrics

For task metrics, we measured participants score after each task session. We also tracked the number of mistakes made during each session. We measured the number of times participants opened and closed applications, as well as the average time they spent using each application. For layerable modes, we calculated the average number of application layers open on a per-frame basis.

To measure the usability of each mode, we employed the System Usability Scale [213], which was part of the post-task questionnaires. We also employed a single ease question in the post-task questionnaire to assess how difficult users found the task under each mode.

To measure spatial awareness, we employed an object recognition quiz as well as an object placement quiz. We used the same testing and scoring methodology as Suma et al. [214] in their previous work evaluating cognitive effects of exploration in mixed reality spaces. For the object recognition quiz, participants were given a list of 30 objects, with half of the objects actually being present in the experiment space, and the other half being absent. Participants were asked to answer true or false for each object. The number of false positives was subtracted from the number of true positives, yielding a score between 0 and 15. Following the object recall quiz, participants took an object placement quiz in which they were given the correct list of 15 present objects, and asked to mark their locations on a 2D top-down view floor plan of the space to the best of their memory. Participants were scored based on the number of objects that were correctly placed relative to other objects, for a max score of 15.

In our post-study questionnaire we focus on overall preferences. Participants were asked to rank in which mode they felt the most productive, fastest, focused, distracted, spatially aware, or tired. We asked participants which mode they preferred the most, which was the easiest to use, and which was the most enjoyable.

7.5 Results

We examine differences between users in Layerable and Immersive modes, with a focus on evaluating Layerable apps in the context of personal computing. Additionally during the course of piloting the user study, we also noticed a trend where users who had prior AR experience tended to score higher overall. Recognizing the importance of application switching in the context of productivity tasks, we decided to examine differences between experienced AR users vs. novice users. Expertise was determined based on subjective responses from the pre-study questionnaire, where participants were asked on a scale of 1 to 5 how familiar they were with Augmented Reality. Those who answered 4 or 5 were considered as experienced AR users. Using this criteria resulted in 23 users categorized as experienced and 21 as novice.

7.5.1 Tutorial Adequacy and HoloLens Usability

We assessed the suitability of our training procedures in post-hoc questionnaires employing a 7-point Likert scale (higher numbers indicating higher amounts of understanding of the tutorial, the task, and ease of use of the HoloLens). 86.4% of participants rated highly (5 or higher) for tutorial understanding, 95.5% of users rated highly for task understanding, and 97.7% of participants rated highly for ease of use.

7.5.2 Task Performance

Participants were scored based on the number of codes they were able to find and enter successfully within a 7-minute task session. We averaged scores for both Immersive and Lay-

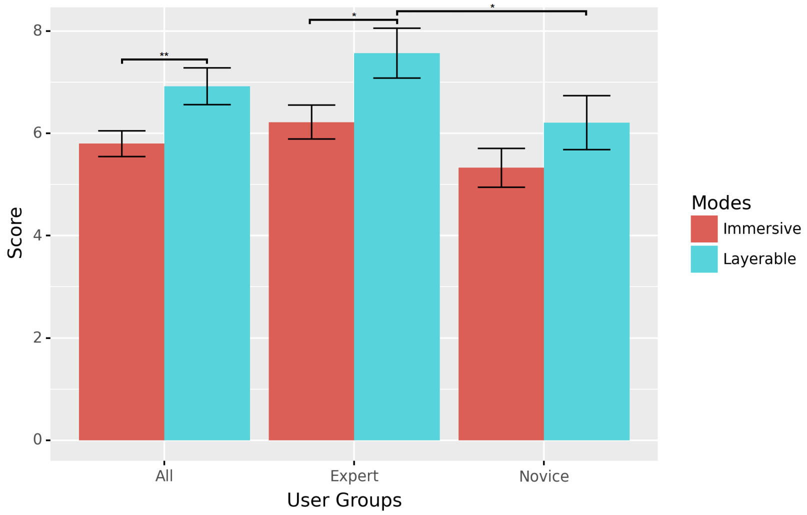


Figure 7.2: Average task score in terms of number of codes successfully found. These are shown according to user group and application mode.

erable modes for all participants, as well as for the subgroups of Expert and Novice, shown in Figure 7.2. We compared scores between modes using the Wilcoxon signed-rank test with Bonferroni correction to account for multiple comparisons error. We report effect size r adopting Cohen's classification [215] of small (0.1 to 0.3), medium (0.3 to 0.5) and large (> 0.5) effect sizes. For significance tests, we used α of 0.05.

When looking at all participants as a whole, we found significantly higher scores ($p = .002, r = .365$) when completing the task under Layerable mode, averaging 6.92 ($SD = 3.39$) compared to 5.8 ($SD = 2.39$) under Immersive mode. We also found significance among experienced AR users ($p = .012, r = .425$), with an average score of 7.57 ($SD = 3.29$) compared to 6.22 ($SD = 2.27$) in Immersive. We did not find significance between modes for the novice group ($p = .178, r = .291$), with an average score of 6.21 ($SD = 3.4$) under Layerable and 5.33 ($SD = 2.46$) under Immersive.

We used the Mann-Whitney U test to analyze differences between experienced and novice users. When aggregating across both modes, experienced users scored significantly higher ($p = .005, r = .242$) in the experimental task ($M = 6.89, SD = 2.89$) compared to novice users

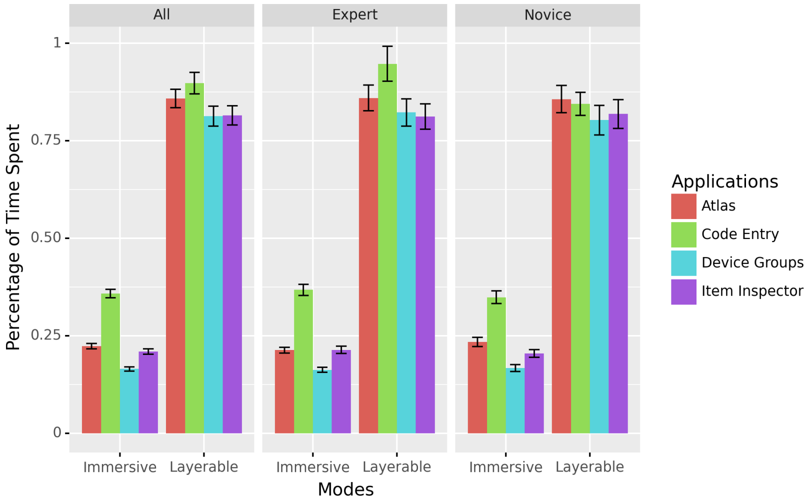


Figure 7.3: Average percentage of time spent in each app. Note that this includes all time that the application was open, not necessarily time spend directly interacting with content.

$(M = 5.77, SD = 2.98)$. They also scored significantly higher $(p = .035, r = .26)$ when using Layerable mode compared to novice users. Results were inconclusive $(p = .068, r = .225)$ when comparing Immersive scores between expert and novice users, with a trend to higher performance by expert users.

We also looked at the number of mistakes participants made during each task session but did not find any significant differences between modes or experience levels. Mistakes were defined as incorrect code entries. On average, participants made 1.59 $(SD = 1.7)$ mistakes with Layerable and 1.61 $(SD = 1.97)$ with Immersive. Experienced AR users averaged 1.78 $(SD = 1.78)$ mistakes compared to 1.4 $(SD = 1.89)$ for novices.

7.5.3 Application Usage

We examined application usage behaviors by looking at time spent in each app as well as app switching actions. We measured the duration of time applications were kept open during a task session. Applications were automatically closed at the end of each task session, so the maximum length of time is seven minutes. Among the total sample population, participants kept apps open significantly longer ($p < .001, r = .867$) in layerable mode ($M = 130, SD = 156.22$), compared to immersive mode ($M = 9.01, SD = 21.58$). While this is unsurprising, it is noteworthy that applicants did not simply leave apps open continuously, as one might expect that to be an optimal strategy. We discuss this further in Section 6.

Additionally, we calculated the average proportion of time spent in each app within a single task session. A breakdown of the mean proportion of time spent in each application during a task session can be found in Figure 7.3. Though we did not find any significant differences, we can see interesting trends in usage that may warrant further exploration. For instance, users appear to engage with context-free applications (Atlas and Code Entry) slightly more often than context-aware applications (Device Groups and Item Inspector). This may be due to users perceiving the space as more cluttered with context-aware applications. We also see that in general, Code Entry is used more often than any other app, with the exception of novice users under layerable mode. This is surprising and may by an indication of choice overload amongst novice users who are inexperienced with the different capabilities of AR.

For application switching behavior, we measured application open and close actions performed by each user. It should be noted that since apps can be layered and open simultaneously, opening an application does not necessarily mean a user has switched their attention to that application's contents. The average number of application open actions in layerable mode were 21.3 ($SD = 12.96$) and 90.34 ($SD = 26.19$) in immersive mode. A Wilcoxon Signed-Rank test showed these to be significantly different ($p < .001, r = .859$). Average counts for the close app action were 14.8 ($SD = 13.49$) for layerable and 88.64 ($SD = 26.1$) for immersive modes. We found these means to also be be significantly different ($p < 0.001, r = .851$).

When comparing between user groups, we did not find differences between experts and novices on the amount of open actions in layerable ($p = .541$) or immersive ($p = .125$), nor did we find differences for close actions in layerable ($p = .663$) or immersive modes ($p = .126$) modes.

Specifically for layerable mode, we were interested in the number of applications participants kept open at any given time. We averaged the number of apps open on a per frame basis for each task session when using layerable mode. Mean apps open was 3.39 ($SD = .694$), with that number being slightly higher amongst AR experts ($M = 3.43, SD = .610$) and slightly lower ($M = 3.35, SD = .781$) amongst AR novices. We wanted to know if participants cycled between different applications or if they chose to keep all applications open simultaneously to provide themselves with the most available information for completing the task. We used a one-sided Wilcoxon Signed-Rank test, subtracting the expected mean of 4 from all our sample observations to fit the test hypothesis. We found average apps open to be significantly less ($p < .001$) than our expected mean, suggesting users don't opt to use all apps simultaneously even though that is, in our opinion, the more efficient strategy.

7.5.4 Usability

We employed the System Usabiliy Scale (SUS) [213] after the first use of each mode, as well as a Single Ease Question (SEQ) rating the ease of task completion from 1 (easy) to 7 (difficult). A Wilcoxon Signed-Rank test showed no differences in SUS score ($p = .607$) between modes with a mean layerable mode score of 67.27 ($SD = 13.91$) and mean immersive mode score of 68.47 ($SD = 15.58$). There were also no differences found between modes when examining the scores of the experienced user groups ($p = .425$) and novice user groups ($p = .708$). When interpreting SUS scores, an 'OK' score is generally 51-71 and a 'Good' score is generally 72-85, so both layerable and immersive modes fall somewhere between 'OK' and 'Good' [216]. We also found no differences in SEQ score between modes ($p = .549$), nor amongst experienced users ($p = .773$) or novices ($p = .598$).

7.5.5 Spatial Awareness

We tested the effects of each mode on spatial awareness with an object recognition and object placement quiz. Quizzes were administered after the first task session only, as we did not want to influence task performance by having participants divert attention to memorizing parts of the space in later trials. As we counterbalanced the starting modes for each participant, we can effectively treat these results as coming from independent groups, but with spatial awareness results for only n=22 (half our user population) participants for each mode. We use a Two-Way ANOVA to analyze the quiz scores, and confirmed normality using Shapiro-Wilk test as well as homogeneity of variances using Levene's test. We used system mode (layerable vs. immersive) and AR experience (expert vs. novice) as our independent factors, using quiz score as our dependent variable for the ANOVA model. Post-hoc analysis was performed using Tukey's HSD test for all pairwise comparisons.

Table 7.2: Two-Way ANOVA of Object Placement Scores

effect	sum sq	df	F	PR(>F)
Mode	50.62	1	5.64	**0.022**
Experience	16.97	1	1.89	0.177
Mode x Experience	8.58	1	0.96	0.334
Residual	358.99	40		

Both quizzes had a max score of 15. Please refer to section 4.2 for details on how the quizzes were scored. In the object placement quiz, we found a significant main effect of system mode on the object score ($p = .026, r = .665$), with users averaging a score of 8.64 ($SD = 2.5$) in layerable compared to 6.55 ($SD = 3.47$) in immersive. We did not find any effect for user experience level ($p = .212$), nor did we find any significant interaction effects between mode used and user experience ($p = .334$). Summary statistics for the ANOVA model are shown in Table 7.2.

For the object recognition quiz, we did not find any statistical significance for either sys-

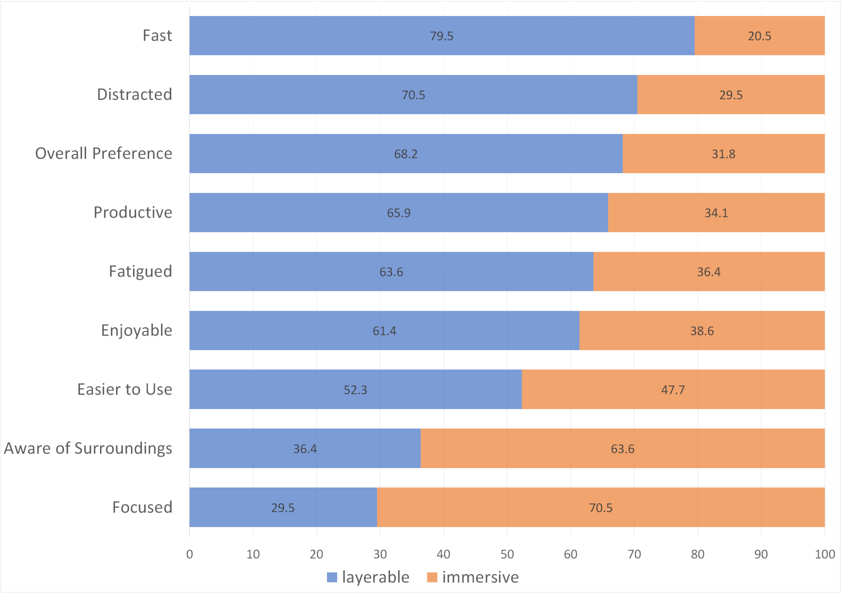


Figure 7.4: Proportion of User's Mode Preferences

tem mode ($p = .497$) or experience level ($p = .245$), nor did we find any significance for the interaction between independent factors ($p = .692$). The average score was 6.5 ($SD = 3.57$) for layerable and 5.73 ($SD = 3.87$) for immersive.

The significantly higher object placement score when using layerable is notable, as in the next section we will see that most users rated themselves as more aware of their surroundings in the immersive mode rather than the layerable mode. While immersive mode may give the feeling of greater spatial awareness due to increased visibility of the physical scene, users' actual spatial awareness performance may be better facilitated by the increased context-related content and visual stimuli in the layerable mode. Similar results have been found in other works regarding AR and memory [70].

7.5.6 Overall Preferences

In the post-study questionnaire, we asked users to rank their preferred modes based on several different criteria, including which mode they felt faster, more productive, more distracted, more fatigued, more focused, and more aware of their surroundings in. We also asked which mode users enjoyed the most, found easiest to use, and preferred overall. Figure 7.4 shows the proportion of user responses for each ranking criteria.

Concerning attitudes around multitasking, 79.5% of users felt faster in layerable, and 65.9% of users felt more productive in layerable. These self-reported rankings fall in line with our task performance results, suggesting that from a task efficiency standpoint, layerable appears to be better.

However, a majority (63.6%) of users also found layerable more tiring to use. We had designed layerable apps with the goal of reducing context switching fatigue, but that does not appear to be the outcome. We believe that while context switching fatigue may be reduced compared to immersive mode, overall fatigue is increased due to increased visual demands or eye strain.

Only 36.4% of users felt more "aware of their surroundings" in layerable and only 29.5% of users felt more focused. 70.5% of users ranked layerable as the more distracting option. These results are counter-intuitive considering spatial awareness quiz scores were generally higher and in some cases significantly higher for layerable mode. We believe these results are due to participants attributing the quality of being "aware of surroundings" to their visibility of the real world. When self-reporting on their spatial understanding, users appear to be biased towards consciously perceived visual cues, which may not be indicative of their actual spatial understanding. Even in AR/VR settings [217,218], spatial awareness is additionally facilitated by other unconscious non-visual inferences such as orientation processing.

7.6 Discussion

Reviewing our initial research questions, our results show Layerable Applications to be a promising application model for Augmented Reality. Layerable was ranked as the more preferred mode to use and was also ranked as more enjoyable by the majority of users in our study.

Statistical analysis showed significant improvements in performance on our tasks (which necessitated cross-referencing) when using layerable applications, compared to a traditional single-application model, on average higher by 1.12 points. We also found significant improvements on layerable apps task performance for experienced AR users compared to novice users, suggesting its suitability as an application paradigm for ‘power users’ who have more technical knowledge or are willing to overcome the initial learning curve. We designed Layerable Apps to increase the degree of control users have on the augmented world. One of our main research questions was to determine user preferences around augmentation control, as such information could be used to inform future application designs. We were concerned due to the nature of the task that users would open all apps all the time, but that was not the case. Rather, our results show that users do frequently choose to switch between applications in layerable mode, switching apps an average of 21.3 times and using app instances an average of 130 seconds.

We found that the number of apps they kept open at any given time was significantly less than the total number of apps available, even though opening all apps may have provided a potentially faster pathway for the task (if one were to discount negative effects from clutter and information overload). It looks like users self-regulated the amount of information display they were willing to take in at a time, shielding against higher levels of clutter and information overload. Additionally, we found some evidence of users being more spatially aware in layerable AR. When analyzing object placement scores, where users had to position objects on a 2D floor plan of the experiment space, those who started in layerable scoring significantly higher. However in contrast to that result, participants also ranked layerable as causing them to be less ‘aware of their surroundings’. These results are interesting, and more work needs to be done to find the ‘sweet spot’ of number of applications and degree of augmentation that users prefer to use.

7.7 Future Work

This work focuses on a simple implementation of the Layerable Apps paradigm, where the onus on view management is strictly on the user. While this form may be appealing to power users and early adopters, it may not be appropriate for mass adoption. In future iterations, we would like to explore how to incorporate view management [152, 154, 181] and information filtering [219, 220] as an element, while preserving the degree of user control that helps distinguish Layerable Apps from other application paradigms. For instance, it may be possible to define a standard set of rules for the presentation and layout of AR app elements, similar to HTML and CSS for web design. Such a system could alleviate the issues of visual fatigue while maintaining the productivity benefits of Layerable Apps.

There is also a mental load involved in determining which applications are appropriate to use in which context and a related challenge for app developers in testing their application to work well in a variety of contexts, as reported in recent developer surveys [61]. One potential solution that we would like to incorporate into Layerable Apps is the inclusion of a "target scene description" with each application, indicating the types of spaces that are appropriate for the application, perhaps in the form of a hierarchical description of objects and surfaces in the scene or similar spatial representation structure. This would provide the user with a quantifiable indicator of how appropriate an application is based on how closely their current space matches the target scene description. Developers would also benefit by being able to narrowly scope their application's operational context and having concrete test cases that they could evaluate their app on.

7.8 Conclusion

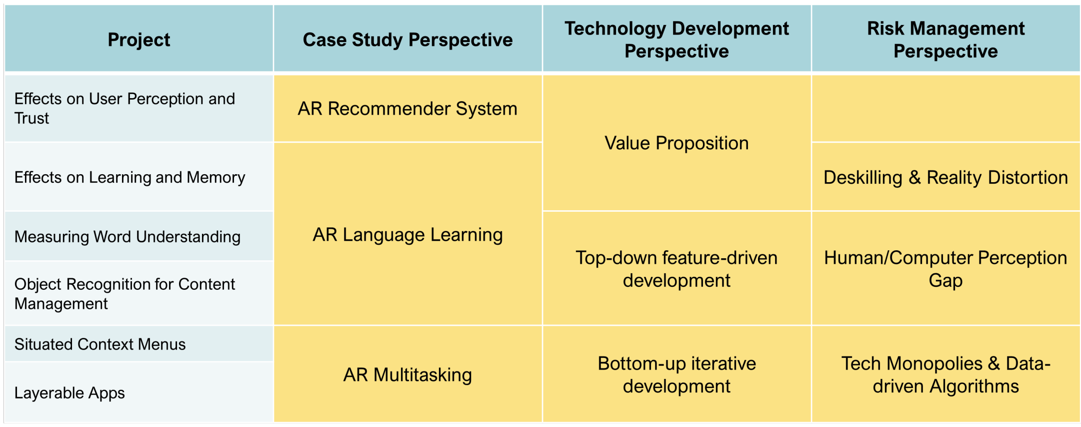
Simultaneous usage of multiple applications in Augmented Reality is a challenging but important problem to solve. In this work, we set forth and evaluated one application model that supports concurrent display of application content, which we call Layerable Apps. We compared a prototype implementation against the commonly used single-application display paradigm through a within-subjects user study with 44 participants. We found significantly higher task performance and demonstrated spatial awareness when using layerable apps, and a majority of users preferred this mode overall. We also analyzed results between experienced and novice AR users and found that experienced users had significantly higher task performance in Layerable as well, suggesting an additional benefit of the system for 'power users.' We documented our design process for the system prototype, experiment task, and choice of sample applications, and analyzed application usage during the study to provide insight towards the design of future multi-app AR interfaces.

Our Layerable Apps is an alternative application model that focuses on smaller scale apps with separate and isolated functionality. Our results showed that users may not actually appreciate having the amount of augmentation predetermined, and enjoy the ability to toggle on and off application layers at will. This is encouraging, and hopefully can be taken as a sign by future platform makers to enable more multitasking capabilities. Smaller scale applications also naturally means less data in a single location. In a tech landscape where data breaches are extremely commonplace, even at some of the largest companies in the world, we should be wary of any AR system that asks us to place all our data in one basket. Our Layerable Apps work shows some benefits of an app model with smaller scale apps, hopefully encouraging AR practitioners to pay greater consideration to these concepts in future projects.

Chapter 8

Conclusion

This dissertation contributes to the growing body of academic work that considers Augmented Reality as a potential new personal or domestic computing interface, exploring ways to improve the design of AR systems and applications in anticipation of a pervasive AR future. We define pervasive AR as the persistent availability of an always-on, always-connected, and always-sensing augmented reality wearable device, most likely a headset. We discussed the societal impact of such a device should it become the de facto consumer computing medium. There are many pros, such as the potential for increased human capabilities and convenient access to information, as well as many potential risks, such as, deskilling, friction between real and virtual environments, destruction of privacy, and monopolization of attention. The contributions in this dissertation are initially structured as a series of investigative research questions, answering the questions of why AR is beneficial compared to other computing mediums, what additional inputs we might want to incorporate into AR systems, and what advancements we can and should implement to AR systems right now. However, the contributions of this dissertation can also be analyzed from other research perspectives as well. In the following sections, we contribute additional meta-analyses into how the work in this dissertation also contributes to the field through case studies, technological development, and risk management. A graphic



Project	Case Study Perspective	Technology Development Perspective	Risk Management Perspective
Effects on User Perception and Trust	AR Recommender System	Value Proposition	
Effects on Learning and Memory	AR Language Learning		Deskilling & Reality Distortion
Measuring Word Understanding		Top-down feature-driven development	Human/Computer Perception Gap
Object Recognition for Content Management			
Situated Context Menus	AR Multitasking	Bottom-up iterative development	Tech Monopolies & Data-driven Algorithms
Layerable Apps			

Figure 8.1: Graphic showing how the chapters of this dissertation can be additionally analyzed from different perspectives.

summarizing how each chapter is used for the additional analyses can be seen in Figure 8.1.

8.1 Case Study Perspective

The contributions of this dissertation can be viewed as a series of case studies that analyze different aspects of pervasive AR. In total, three distinct AR use cases were explored: AR Recommender Systems, AR Language Learning, and AR Multitasking.

Our work on AR recommender systems provides a case study on proactive interfaces, also called implicit interfaces or noncommand interfaces [75]. Proactive interfaces make decisions on behalf of the user, rather than requiring the user to explicitly command the interface through user input. While proactive systems are a theorized component of future pervasive AR, their usage has been underexplored as they are often difficult to implement. Our work provides one of the first implementations of a simple proactive system in AR, and demonstrate how the situated qualities of AR presentations can improve users perception of the recommender system itself.

The AR language learning system is the largest case study in this dissertation, as we explore it throughout multiple research projects. This use case provides an example of a necessarily continuous application. Pervasive AR systems ask us to consider how AR applications might be used in an always-on AR scenario, everyday and throughout different situations in our lives. Such continuous applications are difficult to design, difficult to implement, and difficult to study. In this dissertation, we undertake that difficult work by breaking it up into individual components. The resultant projects, when considered in their totality, inform designers on how to create applications that utilize continuous and recurring interactions.

This dissertation showed different ways to perform AR multitasking. Our two projects, situated context menus and layerable apps, provided insights towards the design of application switching techniques. Pervasive AR places an emphasis on multi-purpose use of AR rather than single-purpose or static usage. Yet there are few AR systems that are intended for personal computing use. Our AR multitasking projects can also be seen as examples of personal computing use cases, and the results of those projects may be more transferable when designing future AR personal computing interfaces.

8.2 Technology Development Perspective

The work presented in this dissertation can also be viewed from the perspective of increasing the progress of technology development, by addressing near-term technical barriers to pervasive AR. Within this perspective, we can analyze our contributions as part of top-down and bottom-up approaches to feature development and system design.

We use a top-down approach to identify existing technical challenges, by conceptualizing a theoretical use case, AR language learning, and working backwards to determine the missing pieces. From there, we focus our efforts on how to design those pieces, and how to integrate them into a coherent system architecture. This process allowed us to demonstrate the potential of tools such as word understanding, object-driven view management, and spatial arrangement, in always-on AR scenarios, and provide some insights on how to effectively design and incorporate said features into an AR application.

From a bottom-up approach, we evaluated the limitations of current AR systems, and identified the application model as a significant limitation to the effective development of pervasive AR interaction techniques. Our work contributes to the development progress of multi-application interaction through the design and evaluation of two multitasking prototypes. In Situated Context Menus, we explored the balance between context-aware application recommendations and user agency. The evaluation provides insights into the design of context-dependent functionality, and how it can be useful to link said functionality to the context source, for instance an object in the environment. In Layerable AR, we explored the degree of augmentation that users prefer. Our results revealed notable insights that can be used by future designers. For instance, despite being a more optimal strategy, users did not layer all applications concurrently. This suggests that there exists a limit to the amount of augmentation that is acceptable to users, which is supported by other prior works related to visual clutter. Our work also examined differences in novice and experienced users, suggesting that for more experienced users that visual clutter threshold may be higher.

In addition to top-down and bottom-up approaches, our work also contributed knowledge about the value-proposition of AR to users and developers through our studies on the effects of AR on user perception, trust, learning, and memory.

8.3 Risk Management Perspective

The goal of this risk management analysis is to provide a long-term perspective of our work. Technology can change quickly, and there is a significant chance that many of our contributions could become irrelevant in just a few years. Move fast and break things' has been the battle cry of the tech industry for the last decade after all [44]. Unfortunately, short-term thinking often comes with unintended consequences. It is useful to consider our work not just in the immediate context, but also 10, or 20 years from now.

In the introduction, we predicted several potential consequences of a pervasive AR future. The first was deskilling and reality distortion. Deskilling refers to the over-reliance on technology as a replacement for existing human skills and capabilities. We posited that the increased convenience of information access would cause greater deskilling to the detriment of society. Reality distortion is a related phenomenon, referring to the potential for an always-on AR system to distort our perception of reality, causing us to overestimate our abilities or worse, to disregard the true nature of the world in favor of the one that is seen through our AR display. Our work attempts to address this risk in two different ways. First, we conducted user studies to identify differences differences in user perception and other cognitive factors between AR and other existing mediums. Through this work, we discovered that AR was not always the preferred modality to accomplish a given task. Instead, it has certain advantages such as the ability to spatially arrange information to improve retention or to conveniently compare items side-by-side. This helps to establish a “right tool for the right task” mindset. To build on that, we showed how to design an application that focuses on improving users skills, using the unique advantages of AR, while still providing value to the user when you take away the AR device. That application was our pervasive AR language learning concept.

The second risk we identified was the context awareness gap, referring to the differences between how humans and computers process semantics and context. This gap, we posited, could lead to interfaces that are high friction and could distract the user from the real world. To address this challenge, we evaluate task-dependant semantic understanding techniques for our AR language learning concept. Our results demonstrate the feasibility of using eye tracking and object recognition to improve the relevant context awareness capabilities of current AR systems, increasing the dynamism of our application and potentially enabling it to adapt more seamlessly to daily life scenarios. This research also provided insights into the deployment and integration of large machine learning models in AR applications. It suggests the need to focus ML efforts on targeted tasks, such as content arrangement, given the large disparity in computational capabilities between AR headset and the GPU servers that ML algorithms are typically developed on.

Finally, we focus on the problem of monopolization and the deployment of data-driven algorithms. These are ongoing socio-technical challenges of today that may very well continue on in the AR future. In this case, data-driven algorithms refer to the use of user data to drive the functionality of a user-facing system or interface, a common practice for large tech companies, particularly social media companies. The deployment of these algorithms has been linked to increased political division, the proliferation of fake news, and increasing distrust in online discourse. Many of the same companies are actively investing in augmented reality, vying to be the first company to have access to the vast multitudes of user data that could be collected from an always-on, wearable AR device. If they were to deploy the same algorithms, we may be looking towards an AR future rife with attention grabbing advertisements, technology overreach, and the deterioration of our personal data privacy and security.

To address this, our work provides alternatives to currently proposed application models. Instead of the large all-encompassing applications that use AI to proactively determine functionality, we designed and prototyped alternative application models, and evaluated them with a focus on multitasking performance. Our systems tackle the issue of privacy through two techniques: abstraction and segmentation.

The first system we designed, situated context menus, provided a balance between using context data for information filtering while still allowing the user to choose for themselves whether or not they want to use the available applications. Situated context menus is an example of abstracting intimate data, in this case the user's physical environment, to the operating system level. Abstraction is one possible solution to improve data privacy, by providing different levels of privacy guarantees at different layers of the technology stack. In the future, we could for instance guarantee that data used by the operating system, such as room information, is encrypted and not shared to other apps or distributed over the network.

The second system we designed, Layerable Apps, helps to improve data privacy by segmenting and separating who holds the data. As AR becomes more widespread, we need to be careful to manage who holds what forms of personal data. Layerable apps is an application model that encourages smaller-scale applications that are service-oriented, in that each app provides only one service or function. This is due in part to our application taxonomy, which treats all forms of apps equally and allows any app, big or small, to render on top of something else. Instead of trapping users in a single app, developers will now have to compete for attention. As long as apps are competing with each other, they have no incentive to share data, making sure no one entity holds all our data.

8.4 Future Directions

My journey as a researcher started with a desire to get in on the ground floor of new and exciting technology, such as AR and machine learning. Today I realize it can be dangerous to bandwagon onto technology trends without thinking of the ethical and long-term considerations. I am more wary than ever about the role technology, and especially large technology companies, have in our future. Looking forward, I want to have a stronger influence on the direction this field takes, as I want to guide it towards more ethical and responsible uses of technology. As the industry pushes forward, I see promising opportunities in the creation of development tools and development platforms for AR applications, namely in the following areas.

8.4.1 Testing Environments and Scene Descriptors

Currently, developers have little to no tools to test their applications within a physical environment, let alone within all the possible environments that their users may encounter. Developers have been vocal about the lack of simulation tools and proper guidelines for making their content fit in any given space [61]. This dissertation provided several solutions to the dynamic adaption of content to space, and researchers continue to explore this field. But general view management techniques may not be enough.

Instead, we might want to look at tools that enable AR apps to target specific spaces. For instance, a developer might want to make an application that only works at certain retail stores, or only works at home, or only works in the living room. If they can constrain their apps to specific types of rooms, they might be able to design for that type of space better. For instance, they could expect that a living room usually contains a TV or a couch, and could design the placement of AR content around those objects. Unfortunately, the only way to do with today's tools is by through manual specification from the user.

A better solution would be to have a standard and generic way to describe scenes. That way, developers could program their application to work on a specific set of scene descriptions. A testing framework could then be created, likely in VR or a game engine, enabling developers to create different unit tests based on representative scenes for each scene descriptor. Operating systems for AR platforms could also use this information to indicate whether applications will work in a given scene (and possibly notify the users as to why it won't work).

Scene graphs are one possible solution for implementing such a scene descriptor. They are commonly used in game development and 3D graphics to represent the positional relationships between objects in a virtual scene. This allows objects, such as a camera, to move itself relative to another object, say a character. However, they can also be used to represent logical, directional, and other forms of semantic relationships. For instance, in the living room example, we could describe that scene as a graph with a root node that represents the room, and two child nodes representing sofa and TV, each connected with the relationship "contains". Scene graphs have two desirable properties we want in a scene description. For one, they can be directly compared through graph matching algorithms [221]. This allows us to provide a quantitative metric for how close (or far) away a scene description is from the current environment, allowing flexibility of scene targeting. Instead of just binary targeting, a developer could warn the user that their current environment is close to their scene descriptions, and may still allow them to use the app with certain caveats. Scene graphs can also be hierarchical, allowing developers to be as specific or as general as they want. In the case of a retail store for instance, an AR app developer might want to ensure a very specific set of target scenes in order to make sure the app is never used except at a particular company's stores. Scene graphs allow the flexibility to do that.

8.4.2 Interoperability and Open Standards

Another component of the developer ecosystem that is necessary is an interoperability standard for AR content. Standards like OpenXR only provide low-level access to underlying display and input hardware. What is necessary for AR applications to run concurrently, and possibly display concurrently, is something more akin to a layout standard and rendering engine, such as those used in web browsers. This would allow the structure of an application's visual content to be communicated to the operating system, providing information for the OS to assist in displaying the content coherently with respect to other open applications.

Similar to the use of HTML on websites, a layout standard would allow for "soft failures" when designing an application, enabling faster iteration and ease of content authoring and creation. For instance, because there is a standard way to present certain types of HTML elements, when developers use those elements in the design of their website, it is unlikely to break the website entirely. Instead, the element will simply default to the standard way of presentation based on where it is in the current DOM hierarchy. An AR layout engine could be built in a similar way. For instance, if there is one application open that presents a 3D model, that may default to being placed in the center. If then the user opens another AR application, instead of also displaying that in the center, the default may be to place them on the left and right of the users field of view, to ensure the contents do not overlap. This ease of authoring helps reduce the barrier to entry for AR designers and keeps app developers on an even playing field as they must use the same standard to communicate the presentation of their applications.

8.4.3 Privacy and Security Guarantees

Lastly, another direction that I see promise in is the implementation of privacy and security guarantees for data generated through AR. This can take the form of system design, policy design, or interface design. For instance, an AR platform could provide OS-level implementations of context-sensing algorithms, abstracting it for the developer to use. The design of their API can be built in a black-box manner so that the developer cannot directly inspect the raw sensor data used, but can only utilize aggregate results, providing safeguards against developer abuse. This is already used in some platforms. The HoloLens 2 includes eye tracking cameras, however the only data that is provided is gaze information. The platform does not expose eye camera images, since they may be used to circumvent biometric authentication systems that utilize retinal or iris scans. However, more work can be done in this space, such as ensuring the data is anonymized and encrypted, guaranteeing that the data can only be used on the device and not wirelessly transmitted, and providing limits to frequency of access.

While systems demonstrating privacy and security guarantees are valuable, many of these issues cannot be tackled without changes to policy. Government policies ensuring the protection of user data are uncommon, but the situation is slowly starting to change. The European Union recently implemented the General Data Protection Regulation (GDPR) law to increase data protection and privacy [222], becoming a model for similar laws in several other countries as well. The law enhances user's control and rights over their personal data, including how it is transferred between EU and EEA areas. In the future, we might need to consider laws that regulate how data is transferred in and out of an AR system, as these devices become more prevalent in our lives.

Interface design is also a meaningful next step to ensuring data privacy. This dissertation has talked about enabling trade-offs between privacy and AR goals. Different users may have different concerns around privacy, and some may be happy to share more data for the sake of convenience or improved functionality. In those instances, we should provide interfaces that effectively communicate the amount of data that is being shared, when it is being shared, to whom it is being shared with, for what purposes it is being used for, and what functionality the user is receiving in return. By offering these tools, we can promote app designs that focus on privacy, and may influence the development of AR towards openness and transparency.

8.5 Discussion

This dissertation contributes research in the growing field of augmented reality, with an emphasis on the design and evaluation of pervasive AR systems. Our research makes use of speculative design, rapid prototyping, user studies, and statistical evaluation, presented through three investigative research questions centered around: (1) The benefits of AR, (2) Additional inputs for AR, and (3) The evolution of existing systems to pervasive systems.

In addition to this investigation, we also contribute three meta-analyses of the presented research. We analyzed the work as a collection of case studies on AR recommender systems, AR language learning, and AR multitasking. We also analyzed our work from the perspective of systems development and iterative design, categorizing our work into top-down and bottom-up approaches to system design. And we analyzed our contributions with a long-term view directed at mitigating the potential risks and consequences of a pervasive AR future, addressing topics we discussed in the introduction.

If you take anything away from this dissertation it should be this: get involved and participate. This field needs more active participation from researchers in influencing the growth and development of pervasive AR and its perception to the public. One way this can be done is by focusing on the design of applications that bring positive benefits even when the AR device is not in use. Another way is by creating tools, systems, and designs, that can be easily deployed, and whose values is quickly demonstrated, such as with Layerable Apps. Augmented Reality has the power to change the world, much like the PC, Internet, and Smartphone before it. We need more researchers not only to demonstrate the capabilities of this technology, but also to understand the pitfalls and help to guide us away from them.

Bibliography

[1] K. Kim, M. Billinghurst, G. Bruder, H. B.-L. Duh, and G. F. Welch, *Revisiting trends in augmented reality research: A review of the 2nd decade of ismar (2008–2017)*, *IEEE transactions on visualization and computer graphics* **24** (2018), no. 11 2947–2962.

[2] J. Grubert, T. Langlotz, S. Zollmann, and H. Regenbrecht, *Towards pervasive augmented reality: Context-awareness in augmented reality*, *IEEE transactions on visualization and computer graphics* **23** (2016), no. 6 1706–1724.

[3] F. Zhou, H. B.-L. Duh, and M. Billinghurst, *Trends in augmented reality tracking, interaction and display: A review of ten years of ismar*, in *2008 7th IEEE/ACM International Symposium on Mixed and Augmented Reality*, pp. 193–202, IEEE, 2008.

[4] M. Billinghurst, A. Clark, G. Lee, *et. al.*, *A survey of augmented reality*, *Foundations and Trends® in Human–Computer Interaction* **8** (2015), no. 2-3 73–272.

[5] L.-H. Lee, T. Braud, P. Zhou, L. Wang, D. Xu, Z. Lin, A. Kumar, C. Bermejo, and P. Hui, *All one needs to know about metaverse: A complete survey on technological singularity, virtual ecosystem, and research agenda*, *arXiv preprint arXiv:2110.05352* (2021).

[6] R. J. Jacob, A. Girouard, L. M. Hirshfield, M. S. Horn, O. Shaer, E. T. Solovey, and J. Zigelbaum, *Reality-based interaction: a framework for post-wimp interfaces*, in *Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 201–210, 2008.

[7] M. Knecht, C. Traxler, O. Mattausch, W. Purgathofer, and M. Wimmer, *Differential instant radiosity for mixed reality*, in *2010 IEEE international symposium on mixed and augmented reality*, pp. 99–107, IEEE, 2010.

[8] P. Kan and H. Kaufmann, *High-quality reflections, refractions, and caustics in augmented reality and their contribution to visual coherence*, in *2012 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pp. 99–108, IEEE, 2012.

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

[10] 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.

[11] F. Lu, S. Davari, L. Lisle, Y. Li, and D. A. Bowman, *Glanceable ar: Evaluating information access methods for head-worn augmented reality*, in *2020 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pp. 930–939, IEEE, 2020.

[12] S. J. Henderson and S. Feiner, *Evaluating the benefits of augmented reality for task localization in maintenance of an armored personnel carrier turret*, in *2009 8th IEEE International Symposium on Mixed and Augmented Reality*, pp. 135–144, IEEE, 2009.

[13] S. J. Henderson and S. K. Feiner, *Augmented reality in the psychomotor phase of a procedural task*, in *2011 10th IEEE international symposium on mixed and augmented reality*, pp. 191–200, IEEE, 2011.

[14] P. C. Thomas and W. David, *Augmented reality: An application of heads-up display technology to manual manufacturing processes*, in *Hawaii international conference on system sciences*, vol. 2, ACM SIGCHI Bulletin, 1992.

[15] D. Curtis, D. Mizell, P. Gruenbaum, and A. Janin, *Several devils in the details: making an ar application work in the airplane factory*, in *Proc. Int'l Workshop Augmented Reality*, pp. 47–60, 1999.

[16] A. Tang, C. Owen, F. Biocca, and W. Mou, *Comparative effectiveness of augmented reality in object assembly*, in *Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 73–80, 2003.

[17] S. Wiedenmaier, O. Oehme, L. Schmidt, and H. Luczak, *Augmented reality (ar) for assembly processes design and experimental evaluation*, *International journal of Human-Computer interaction* **16** (2003), no. 3 497–514.

[18] L. Hou, X. Wang, L. Bernold, and P. E. Love, *Using animated augmented reality to cognitively guide assembly*, *Journal of Computing in Civil Engineering* **27** (2013), no. 5 439–451.

[19] A. Javornik, *Augmented reality: Research agenda for studying the impact of its media characteristics on consumer behaviour*, *Journal of Retailing and Consumer Services* **30** (2016) 252–261.

[20] G. McLean and A. Wilson, *Shopping in the digital world: Examining customer engagement through augmented reality mobile applications*, *Computers in Human Behavior* **101** (2019) 210–224.

[21] B. Settles and B. Meeder, *A trainable spaced repetition model for language learning*, in *Proceedings of the 54th annual meeting of the association for computational linguistics (volume 1: Long papers)*, pp. 1848–1858, 2016.

[22] C. Thompson, *How khan academy is changing the rules of education*, *Wired magazine* **126** (2011) 1–5.

[23] M. Billinghurst, H. Kato, and I. Poupyrev, *The magicbook: a transitional ar interface*, *Computers & Graphics* **25** (2001), no. 5 745–753.

[24] A. Dünser and E. Hornecker, *An observational study of children interacting with an augmented story book*, in *International Conference on Technologies for E-Learning and Digital Entertainment*, pp. 305–315, Springer, 2007.

[25] A. Dünser, L. Walker, H. Horner, and D. Bentall, *Creating interactive physics education books with augmented reality*, in *Proceedings of the 24th Australian computer-human interaction conference*, pp. 107–114, 2012.

[26] A. MacWilliams, *A decentralized adaptive architecture for ubiquitous augmented reality systems.* PhD thesis, Technische Universität München, 2005.

[27] M. Huber, D. Pustka, P. Keitler, F. Echtler, and G. Klinker, *A system architecture for ubiquitous tracking environments*, in *2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality*, pp. 211–214, IEEE, 2007.

[28] T. Verbelen, T. Stevens, P. Simoens, F. De Turck, and B. Dhoedt, *Dynamic deployment and quality adaptation for mobile augmented reality applications*, *Journal of Systems and Software* **84** (2011), no. 11 1871–1882.

[29] Y. Xu, N. Stojanovic, L. Stojanovic, A. Cabrera, and T. Schuchert, *An approach for using complex event processing for adaptive augmented reality in cultural heritage domain: experience report*, in *Proceedings of the 6th ACM International Conference on Distributed Event-Based Systems*, pp. 139–148, 2012.

[30] A. Dünser, R. Grasset, and H. Farrant, *Towards immersive and adaptive augmented reality exposure treatment*, *Annual Review of Cybertherapy and Telemedicine 2011* (2011) 37–41.

[31] K. Tateno, F. Tombari, I. Laina, and N. Navab, *Cnn-slam: Real-time dense monocular slam with learned depth prediction*, in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 6243–6252, 2017.

[32] M. Runz, M. Buffier, and L. Agapito, *Maskfusion: Real-time recognition, tracking and reconstruction of multiple moving objects*, in *2018 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pp. 10–20, IEEE, 2018.

[33] J. Huang, V. Rathod, C. Sun, M. Zhu, A. Korattikara, A. Fathi, I. Fischer, Z. Wojna, Y. Song, S. Guadarrama, *et. al.*, *Speed/accuracy trade-offs for modern convolutional object detectors*, in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7310–7311, 2017.

[34] D. So, W. Mańke, H. Liu, Z. Dai, N. Shazeer, and Q. V. Le, *Searching for efficient transformers for language modeling*, *Advances in Neural Information Processing Systems* **34** (2021) 6010–6022.

[35] S. Minaee, Y. Y. Boykov, F. Porikli, A. J. Plaza, N. Kehtarnavaz, and D. Terzopoulos, *Image segmentation using deep learning: A survey*, *IEEE transactions on pattern analysis and machine intelligence* (2021).

[36] B. Jacob, S. Kligys, B. Chen, M. Zhu, M. Tang, A. Howard, H. Adam, and D. Kalenichenko, *Quantization and training of neural networks for efficient integer-arithmetic-only inference*, in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2704–2713, 2018.

[37] T. Hoefler, D. Alistarh, T. Ben-Nun, N. Dryden, and A. Peste, *Sparsity in deep learning: Pruning and growth for efficient inference and training in neural networks.*, *J. Mach. Learn. Res.* **22** (2021), no. 241 1–124.

[38] S. J. Nowlan and G. E. Hinton, *Simplifying neural networks by soft weight-sharing*, *Neural Comput.* **4** (1992), no. 4 473–493.

[39] S. Han, H. Mao, and W. J. Dally, *Deep compression: Compressing deep neural network with pruning, trained quantization and huffman coding*, in *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings* (Y. Bengio and Y. LeCun, eds.), 2016.

[40] K. Ullrich, E. Meeds, and M. Welling, *Soft weight-sharing for neural network compression*, in *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*, OpenReview.net, 2017.

[41] S. Feiner, B. MacIntyre, T. Höllerer, and A. Webster, *A touring machine: Prototyping 3d mobile augmented reality systems for exploring the urban environment*, *Personal Technologies* **1** (1997), no. 4 208–217.

[42] C.-X. Wang, M. Di Renzo, S. Stanczak, S. Wang, and E. G. Larsson, *Artificial intelligence enabled wireless networking for 5g and beyond: Recent advances and future challenges*, *IEEE Wireless Communications* **27** (2020), no. 1 16–23.

[43] T. Dybå and T. Dingsøyr, *Empirical studies of agile software development: A systematic review*, *Information and software technology* **50** (2008), no. 9-10 833–859.

[44] J. Taplin, *Move fast and break things: How Facebook, Google, and Amazon have cornered culture and what it means for all of us.* Pan Macmillan, 2017.

[45] A. Quigley and J. Grubert, *Perceptual and social challenges in body proximate display ecosystems*, in *Proceedings of the 17th International Conference on Human-Computer Interaction with Mobile Devices and Services Adjunct*, pp. 1168–1174, 2015.

[46] T. Miyashita, P. Meier, T. Tachikawa, S. Orlic, T. Eble, V. Scholz, A. Gapel, O. Gerl, S. Arnaudov, and S. Lieberknecht, *An augmented reality museum guide*, in *2008 7th IEEE/ACM International Symposium on Mixed and Augmented Reality*, pp. 103–106, IEEE, 2008.

[47] D. Chekhlov, A. P. Gee, A. Calway, and W. Mayol-Cuevas, *Ninja on a plane: Automatic discovery of physical planes for augmented reality using visual slam*, in *2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality*, pp. 153–156, IEEE, 2007.

[48] L. Chen, T. W. Day, W. Tang, and N. W. John, *Recent developments and future challenges in medical mixed reality*, in *2017 IEEE international symposium on mixed and augmented reality (ISMAR)*, pp. 123–135, IEEE, 2017.

[49] S. Di Verdi, D. Nurmi, and T. Höllerer, *Arwin-a desktop augmented reality window manager*, in *The Second IEEE and ACM International Symposium on Mixed and Augmented Reality, 2003. Proceedings.*, pp. 298–299, IEEE, 2003.

[50] S. DiVerdi, T. Höllerer, and R. Schreyer, *Level of detail interfaces*, in *Third IEEE and ACM International Symposium on Mixed and Augmented Reality*, pp. 300–301, IEEE, 2004.

[51] J. Grubert, T. Langlotz, and R. Grasset, *Augmented reality browser survey*, *Institute for computer graphics and vision, University of Technology Graz, technical report* **1101** (2011) 37.

[52] T. Höllerer, S. Feiner, and J. Pavlik, *Situated documentaries: Embedding multimedia presentations in the real world*, in *Digest of Papers. Third International Symposium on Wearable Computers*, pp. 79–86, IEEE, 1999.

[53] R. Kooper and B. MacIntyre, *Browsing the real-world wide web: Maintaining awareness of virtual information in an ar information space*, *International Journal of Human-Computer Interaction* **16** (2003), no. 3 425–446.

[54] D. Harborth and S. Pape, *Exploring the hype: Investigating technology acceptance factors of pokémon go*, in *2017 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pp. 155–168, IEEE, 2017.

[55] J. H. Rah, A. A. Cronin, B. Badgaiyan, V. M. Aguayo, S. Coates, and S. Ahmed, *Household sanitation and personal hygiene practices are associated with child stunting in rural india: a cross-sectional analysis of surveys*, *BMJ open* **5** (2015), no. 2 e005180.

[56] B. Sparrow, J. Liu, and D. M. Wegner, *Google effects on memory: Cognitive consequences of having information at our fingertips*, *Science* **333** (Aug., 2011) 776–778.

[57] H. H. Wilmer, L. E. Sherman, and J. M. Chein, *Smartphones and cognition: A review of research exploring the links between mobile technology habits and cognitive functioning*, Apr., 2017.

[58] E. Pariser, *The filter bubble: What the Internet is hiding from you.* Penguin UK, 2011.

[59] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. A. Efros, *Context encoders: Feature learning by inpainting*, in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2536–2544, 2016.

[60] S. Palmer, *Fundamental aspects of cognitive representation*, .

[61] N. Ashtari, A. Bunt, J. McGrenere, M. Nebeling, and P. K. Chilana, *Creating augmented and virtual reality applications: Current practices, challenges, and opportunities*, in *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, pp. 1–13, 2020.

[62] G. D. Abowd, A. K. Dey, P. J. Brown, N. Davies, M. Smith, and P. Steggles, *Towards a better understanding of context and context-awareness*, in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 1707, pp. 304–307, Springer Verlag, 1999.

[63] M. C. Tacca, *Commonalities between perception and cognition*, *Frontiers in psychology* **2** (2011) 358.

[64] Politico, *Amazon gave Ring videos to police without owners' permission*, 2022 (accessed September 13 2022). `https://www.politico.com/news/2022/07/13/amazon-gave-ring-videos-to-police-without-owners-permission-00045513`.

[65] J. Isaak and M. J. Hanna, *User data privacy: Facebook, cambridge analytica, and privacy protection*, *Computer* **51** (2018), no. 8 56–59.

[66] Statista, *Meta: advertising revenue worldwide 2009-2021*, 2022 (accessed September 13 2022). `https://www.statista.com/statistics/271258/facebooks-advertising-revenue-worldwide/`.

[67] S. Brooks, *Does personal social media usage affect efficiency and well-being?*, *Computers in Human Behavior* **46** (2015) 26–37.

[68] B. Huynh, A. Ibrahim, Y. Chang, T. Höllerer, and J. O'Donovan, *A study of situated product recommendations in augmented reality*, in *2018 IEEE International Conference on Artificial Intelligence and Virtual Reality, AIVR 2018, Taichung, Taiwan, December 10-12, 2018*, pp. 35–43, IEEE Computer Society, 2018.

[69] B. Huynh, A. Ibrahim, Y. Chang, T. Höllerer, and J. O'Donovan, *User perception of situated product recommendations in augmented reality*, *Int. J. Semantic Comput.* **13** (2019), no. 3 289–310.

[70] A. Ibrahim, B. Huynh, J. Downey, T. Höllerer, D. Chun, and J. O'donovan, *Arbis pictus: A study of vocabulary learning with augmented reality*, *IEEE transactions on visualization and computer graphics* **24** (2018), no. 11 2867–2874.

[71] J. Orlosky, B. Huynh, and T. Höllerer, *Using eye tracked virtual reality to classify understanding of vocabulary in recall tasks*, in *2019 IEEE International Conference on Artificial Intelligence and Virtual Reality, AIVR 2019, San Diego, CA, USA, December 9-11, 2019*, pp. 66–73, IEEE, 2019.

[72] B. Huynh, J. Orlosky, and T. Höllerer, *In-situ labeling for augmented reality language learning*, in *IEEE Conference on Virtual Reality and 3D User Interfaces, VR 2019, Osaka, Japan, March 23-27, 2019*, pp. 1606–1611, IEEE, 2019.

[73] B. Huynh, J. Orlosky, and T. Höllerer, *Semantic labeling and object registration for augmented reality language learning*, in *IEEE Conference on Virtual Reality and 3D User Interfaces, VR 2019, Osaka, Japan, March 23-27, 2019*, pp. 986–987, IEEE, 2019.

[74] B. Huynh, J. Orlosky, and T. Höllerer, *Designing a multitasking interface for object-aware AR applications*, in *2020 IEEE International Symposium on Mixed and Augmented Reality Adjunct, ISMAR 2020 Adjunct, Recife, Brazil, November 9-13, 2020*, pp. 39–40, IEEE, 2020.

[75] J. Nielsen, *Noncommand user interfaces*, *Communications of the ACM* **36** (1993), no. 4 82–100.

[76] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl, *Grouplens: An open architecture for collaborative filtering of netnews*, in *Proceedings of the 1994 ACM Conference on Computer Supported Cooperative Work*, CSCW '94, (New York, NY, USA), pp. 175–186, ACM, 1994.

[77] J. L. Herlocker, J. A. Konstan, and J. Riedl, *Explaining collaborative filtering recommendations*, in *ACM conference on Computer supported cooperative work*, pp. 241–250, 2000.

[78] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, *Item-based collaborative filtering recommendation algorithms*, in *Proceedings of the 10th International Conference on World Wide Web*, WWW '01, (New York, NY, USA), pp. 285–295, ACM, 2001.

[79] V. Tamturk, *The-ROI-of-Recommendation-Engines*, 2017 (accessed April 1 2017). `http://bit.ly/2nW2aUz`.

[80] Zacks Equity Research, *Is Apple Looking to Expand in the Augmented Reality World?*, 2017 (accessed April 1 2017). `http://bit.ly/2oTwjBv`.

[81] S. Erickson, *Microsoft HoloLens and Lowe's, working to redefine your next home renovation*, 2017 (accessed April 2 2017). `http://bit.ly/2owhIju`.

[82] M. Riar, J. J. Korbel, N. Xi, R. Zarnekow, and J. Hamari, *The use of augmented reality in retail: a review of literature*, in *Proceedings of the 54th Hawaii International Conference on System Sciences*, p. 638, 2021.

[83] J. Stoyanova, R. Goncalves, A. Coelhc, and P. Brito, *Real-time augmented reality shopping platform for studying consumer cognitive experiences*, in *2013 2nd Experiment@ International Conference (exp.at'13)*, pp. 194–195, IEEE, sep, 2013.

[84] Y. Lu and S. Smith, *Augmented Reality E-Commerce System: A Case Study*, *Journal of Computing and Information Science in Engineering* **10** (2010), no. 2 021005.

[85] T. Olsson, E. Lagerstam, T. Kärkkäinen, and K. Väänänen-Vainio-Mattila, *Expected user experience of mobile augmented reality services: a user study in the context of shopping centres*, *Personal and Ubiquitous Computing* **17** (feb, 2013) 287–304.

[86] B. Wang, M. Ester, J. Bu, and D. Cai, *Who also likes it? generating the most persuasive social explanations in recommender systems*, in *AAAI Conference on Artificial Intelligence*, 2014.

[87] M. Balduini, I. Celino, D. Dell'Aglio, E. D. Valle, Y. Huang, T. Lee, S.-H. Kim, and V. Tresp, *BOTTARI: An augmented reality mobile application to deliver personalized and location-based recommendations by continuous analysis of social media streams*, *Web Semantics: Science, Services and Agents on the World Wide Web* **16** (2012) 33–41.

[88] K. McCarthy, J. Reilly, L. McGinty, and B. Smyth, *Experiments in dynamic critiquing*, in *Proceedings of the 10th International Conference on Intelligent User Interfaces*, IUI '05, (New York, NY, USA), pp. 175–182, ACM, 2005.

[89] N. Tintarev and J. Masthoff, *Effective explanations of recommendations: User-centered design*, in *Proceedings of the 2007 ACM Conference on Recommender Systems*, RecSys '07, (New York, NY, USA), pp. 153–156, ACM, 2007.

[90] N. Tintarev and R. Kutlak, *Explanations - making plans scrutable with argumentation and natural language generation*, in *Intelligent User Interfaces (demo track)*, 2014.

[91] J. O'Donovan, B. Smyth, B. Gretarsson, S. Bostandjiev, and T. Höllerer, *Peerchooser: Visual interactive recommendation*, in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '08, (New York, NY, USA), pp. 1085–1088, ACM, 2008.

[92] K. Verbert, D. Parra, P. Brusilovsky, and E. Duval, *Visualizing recommendations to support exploration, transparency and controllability*, in *International Conference on Intelligent User Interfaces*, IUI '13, pp. 351–362, 2013.

[93] D. Parra, P. Brusilovsky, and C. Trattner, *See what you want to see: Visual user-driven approach for hybrid recommendation*, in *Proceedings of the 19th international conference on Intelligent User Interfaces*, pp. 235–240, ACM, 2014.

[94] S. Bostandjiev, J. O'Donovan, and T. Höllerer, *Linkedvis: exploring social and semantic career recommendations*, in *18th International Conference on Intelligent User Interfaces, IUI 2013, Santa Monica, CA, USA, March 19-22, 2013* (J. Kim, J. Nichols, and P. A. Szekely, eds.), pp. 107–116, ACM, 2013.

[95] J. L. Herlocker, J. A. Konstan, L. Terveen, and J. T. Riedl, *Evaluating collaborative filtering recommender systems*, *ACM Trans. Inf. Syst.* **22** (2004), no. 1 5–53.

[96] J. Schaffer, P. Giridhar, D. Jones, T. Höllerer, T. Abdelzaher, and J. O'Donovan, *Getting the message?: A study of explanation interfaces for microblog data analysis*, in *Intelligent User Interfaces*, IUI '15, pp. 345–356, 2015.

[97] B. P. Knijnenburg, S. Bostandjiev, J. O'Donovan, and A. Kobsa, *Inspectability and control in social recommenders*, in *Conference on Recommender Systems*, RecSys '12, pp. 43–50, 2012.

[98] M. Nilashi, D. Jannach, O. B. Ibrahim, M. D. Esfahani, and H. Ahmadi, *Recommendation quality, transparency, and website quality for trust-building in recommendation agents*, *Electron. Commer. Rec. Appl.* **19** (Sept., 2016) 70–84.

[99] J. O'Donovan, B. Smyth, V. Evrim, and D. McLeod, *Extracting and visualizing trust relationships from online auction feedback comments*, in *IJCAI*, pp. 2826–2831, 2007.

[100] S. Fazeli, B. Loni, A. Bellogin, H. Drachsler, and P. Sloep, *Implicit vs. explicit trust in social matrix factorization*, in *Proceedings of the 8th ACM Conference on Recommender Systems*, RecSys '14, (New York, NY, USA), pp. 317–320, ACM, 2014.

[101] P. Massa and P. Avesani, *Trust-aware recommender systems*, in *Proceedings of the 2007 ACM Conference on Recommender Systems*, RecSys '07, (New York, NY, USA), pp. 17–24, ACM, 2007.

[102] R. Guha, R. Kumar, P. Raghavan, and A. Tomkins, *Propagation of trust and distrust*, in *Proceedings of the 13th International Conference on World Wide Web*, WWW '04, (New York, NY, USA), pp. 403–412, ACM, 2004.

[103] J. L. Harman, J. O'Donovan, T. F. Abdelzaher, and C. Gonzalez, *Dynamics of human trust in recommender systems*, in *RecSys*, pp. 305–308, ACM, 2014.

[104] K. Yu, S. Berkovsky, R. Taib, D. Conway, J. Zhou, and F. Chen, *User trust dynamics: An investigation driven by differences in system performance*, in *Proceedings of the 22Nd International Conference on Intelligent User Interfaces*, IUI '17, (New York, NY, USA), pp. 307–317, ACM, 2017.

[105] S. Nakagawa and H. Schielzeth, *A general and simple method for obtaining r2 from generalized linear mixed-effects models*, *Methods in Ecology and Evolution* **4** (2013), no. 2 133–142.

[106] F. A. Yates, *The Art of Memory*. University of Chicago Press, 1966.

[107] A. Metivier, *How to Learn and Memorize German Vocabulary: ... Using a Memory Palace Specifically Designed for the German Language (and Adaptable to Many Other Languages Too)*. CreateSpace Independent Publishing Platform, 2012.

[108] T. Ishikawa, H. Fujiwara, O. Imai, and A. Okabe, *Wayfinding with a GPS-based mobile navigation system: A comparison with maps and direct experience*, *Journal of Environmental Psychology* **28** (mar, 2008) 74–82.

[109] R. E. Mayer, *The Cambridge handbook of multimedia learning*. Cambridge University Press, 2005.

[110] R. E. Mayer, *Applying the science of learning*. Pearson/Allyn & Bacon Boston, MA, 2011.

[111] R. E. Mayer and V. K. Sims, *For whom is a picture worth a thousand words? extensions of a dual-coding theory of multimedia learning.*, *Journal of educational psychology* **86** (1994), no. 3 389.

[112] D. M. Chun and J. L. Plass, *Effects of multimedia annotations on vocabulary acquisition*, *The modern language journal* **80** (1996), no. 2 183–198.

[113] J. L. Plass, D. M. Chun, R. E. Mayer, and D. Leutner, *Supporting visual and verbal learning preferences in a second-language multimedia learning environment.*, *Journal of educational psychology* **90** (1998), no. 1 25–36.

[114] M. Yoshii, *L1 and l2 glosses: Their effects on incidental vocabulary learning*, *Language learning and technology* **10** (2006), no. 3 85–101.

[115] R. Moreno and R. E. Mayer, *Cognitive principles of multimedia learning: The role of modality and contiguity.*, *Journal of educational psychology* **91** (1999), no. 2 358.

[116] E. Türk and G. Erçetin, *Effects of interactive versus simultaneous display of multimedia glosses on l2 reading comprehension and incidental vocabulary learning*, *Computer Assisted Language Learning* **27** (2014), no. 1 1–25.

[117] G. Culbertson, S. Wang, M. Jung, and E. Andersen, *Social situational language learning through an online 3d game*, in *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, pp. 957–968, ACM, 2016.

[118] C. J. Cai, P. J. Guo, J. Glass, and R. C. Miller, *Wait-learning: leveraging conversational dead time for second language education*, in *CHI'14 Extended Abstracts on Human Factors in Computing Systems*, pp. 2239–2244, ACM, 2014.

[119] O. Scrivner, J. Madewell, C. Buckley, and N. Perez, *Augmented reality digital technologies (ardt) for foreign language teaching and learning*, in *Future Technologies Conference (FTC)*, pp. 395–398, IEEE, 2016.

[120] R. Godwin-Jones, *Augmented reality and language learning: From annotated vocabulary to place-based mobile games*, *Language learning and technology* **20(3)** (2016) 9–19.

[121] Y. Liu, D. Holden, and D. Zheng, *Analyzing students' language learning experience in an augmented reality mobile game: an exploration of an emergent learning environment*, *Procedia-Social and Behavioral Sciences* **228** (2016) 369–374.

[122] T. Nakata, *Computer-assisted second language vocabulary learning in a paired-associate paradigm: A critical investigation of flashcard software*, *Computer Assisted Language Learning* **24** (2011), no. 1 17–38.

[123] R. Grasset, A. Duenser, H. Seichter, and M. Billinghurst, *The mixed reality book: a new multimedia reading experience*, in *CHI'07 extended abstracts on Human factors in computing systems*, pp. 1953–1958, ACM, 2007.

[124] P. Seedhouse, A. Preston, P. Olivier, D. Jackson, P. Heslop, M. Balaam, A. Rafiev, and M. Kipling, *The european digital kitchen project*, *Bellaterra journal of teaching and learning language and literature* **7** (2014), no. 1 0001–16.

[125] M. Dunleavy and C. Dede, *Augmented reality teaching and learning*, in *Handbook of research on educational communications and technology*, pp. 735–745. Springer, 2014.

[126] O. Rosello, M. Exposito, and P. Maes, *NeverMind: Using Augmented Reality for Memorization*, in *Proceedings of the 29th Annual Symposium on User Interface Software and Technology - UIST '16 Adjunct*, (New York, New York, USA), pp. 215–216, ACM Press, 2016.

[127] Y. Fujimoto, G. Yamamoto, T. Taketomi, J. Miyazaki, and H. Kato, *Relationship between features of augmented reality and user memorization*, in *2012 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pp. 279–280, IEEE, Nov., 2012.

[128] M. F. Costabile, A. De Angeli, R. Lanzilotti, C. Ardito, P. Buono, and T. Pederson, *Explore! possibilities and challenges of mobile learning*, in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pp. 145–154, ACM, 2008.

[129] N. Yannier, K. R. Koedinger, and S. E. Hudson, *Learning from mixed-reality games: Is shaking a tablet as effective as physical observation?*, in *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, pp. 1045–1054, ACM, 2015.

[130] H. Kaufmann and D. Schmalstieg, *Mathematics and geometry education with collaborative augmented reality*, *Computers & graphics* **27** (2003), no. 3 339–345.

[131] D. Schmalstieg and D. Wagner, *Experiences with handheld augmented reality*, in *Mixed and Augmented Reality, 2007. ISMAR 2007. 6th IEEE and ACM International Symposium on*, pp. 3–18, IEEE, 2007.

[132] R. Trask, *The History of Basque*. Routledge, 1997.

[133] V. I. Levenshtein, *Binary codes capable of correcting deletions, insertions, and reversals*, .

[134] B. Rhodes and T. Starner, *Remembrance agent: A continuously running automated information retrieval system*, in *The Proceedings of The First International Conference on The Practical Application Of Intelligent Agents and Multi Agent Technology*, pp. 487–495, 1996.

[135] R. Azuma, Y. Baillot, R. Behringer, S. Feiner, S. Julier, and B. MacIntyre, *Recent advances in augmented reality*, *IEEE computer graphics and applications* **21** (2001), no. 6 34–47.

[136] J. M. Henderson, S. V. Shinkareva, J. Wang, S. G. Luke, and J. Olejarczyk, *Predicting cognitive state from eye movements*, *PloS one* **8** (2013), no. 5 e64937.

[137] S. P. Marshall, *Identifying cognitive state from eye metrics*, *Aviation, space, and environmental medicine* **78** (2007), no. 5 B165–B175.

[138] M. K. Eckstein, B. Guerra-Carrillo, A. T. Miller Singley, and S. A. Bunge, *Beyond eye gaze: What else can eyetracking reveal about cognition and cognitive development?*, *Dev. Cogn. Neurosci.* **25** (June, 2017) 69–91.

[139] C. Igel, V. Heidrich-Meisner, and T. Glasmachers, *Shark*, *J. Mach. Learn. Res.* **9** (2008), no. 33 993–996.

[140] J. Karolus, P. W. Wozniak, L. L. Chuang, and A. Schmidt, *Robust gaze features for enabling language proficiency awareness*, in *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, CHI '17, (New York, NY, USA), pp. 2998–3010, Association for Computing Machinery, May, 2017.

[141] R. Godwin-Jones, *Emerging technologies from memory palaces to spacing algorithms: approaches to secondlanguage vocabulary learning*, *Language, Learning & Technology* **14** (2010), no. 2 4–11.

[142] L. Von Ahn, *Duolingo: learn a language for free while helping to translate the web*, in *Proceedings of the 2013 international conference on Intelligent user interfaces*, pp. 1–2, ACM, 2013.

[143] G. H. Bower, *Analysis of a mnemonic device: Modern psychology uncovers the powerful components of an ancient system for improving memory*, *American Scientist* **58** (1970), no. 5 496–510.

[144] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, *Semantic Image Segmentation with Deep Convolutional Nets and Fully Connected CRFs*, arXiv:1412.7062. http://arxiv.org/abs/1412.7062.

[145] C. Ma, J.-B. Huang, X. Yang, and M.-H. Yang, *Hierarchical Convolutional Features for Visual Tracking*, in *2015 IEEE International Conference on Computer Vision (ICCV)*, pp. 3074–3082, IEEE, 2015. arXiv:1707.0381.

[146] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, *You only look once: Unified, real-time object detection*, in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 779–788, 2016.

[147] F. Ma and S. Karaman, *Sparse-to-Dense: Depth Prediction from Sparse Depth Samples and a Single Image*, arXiv:1709.0749. http://arxiv.org/abs/1709.07492.

[148] I. Armeni, S. Sax, A. R. Zamir, and S. Savarese, *Joint 2D-3D-Semantic Data for Indoor Scene Understanding*, arXiv:1702.0110. http://arxiv.org/abs/1702.01105.

[149] A. Dai, A. X. Chang, M. Savva, M. Halber, T. Funkhouser, and M. Nießner, *Scannet: Richly-annotated 3d reconstructions of indoor scenes*, in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5828–5839, 2017.

[150] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, *Pointnet: Deep learning on point sets for 3d classification and segmentation*, in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 652–660, 2017.

[151] K.-C. Lien, B. Nuernberger, T. Höllerer, and M. Turk, *Ppv: Pixel-point-volume segmentation for object referencing in collaborative augmented reality*, in *Mixed and Augmented Reality (ISMAR), 2016 IEEE International Symposium on*, pp. 77–83, IEEE, 2016.

[152] B. Bell, S. Feiner, and T. Höllerer, *View management for virtual and augmented reality*, in *Proceedings of the 14th annual ACM symposium on User interface software and technology*, pp. 101–110, ACM, 2001.

[153] R. Azuma and C. Furmanski, *Evaluating label placement for augmented reality view management*, in *Proceedings of the 2nd IEEE/ACM international Symposium on Mixed and Augmented Reality*, p. 66, IEEE Computer Society, 2003.

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

[155] F. Shibata, H. Nakamoto, R. Sasaki, A. Kimura, and H. Tamura, *A view management method for mobile mixed reality systems.*, in *IPT/EGVE*, pp. 17–24, Citeseer, 2008.

[156] S. Pick, B. Hentschel, I. Tedjo-Palczynski, M. Wolter, and T. Kuhlen, *Automated positioning of annotations in immersive virtual environments*, in *Proceedings of the 16th Eurographics conference on Virtual Environments & Second Joint Virtual Reality*, pp. 1–8, Eurographics Association, 2010.

[157] J. L. Gabbard, J. E. Swan, and D. Hix, *The effects of text drawing styles, background textures, and natural lighting on text legibility in outdoor augmented reality*, *Presence: Teleoperators & Virtual Environments* **15** (2006), no. 1 16–32.

[158] J. L. Gabbard, J. E. Swan, D. Hix, S.-J. Kim, and G. Fitch, *Active text drawing styles for outdoor augmented reality: A user-based study and design implications*, in *Virtual Reality Conference, 2007. VR'07. IEEE*, pp. 35–42, IEEE, 2007.

[159] E. Mendez, S. Feiner, and D. Schmalstieg, *Focus and context in mixed reality by modulating first order salient features*, in *International Symposium on Smart Graphics*, pp. 232–243, Springer, 2010.

[160] L. Siklóssy, *Natural language learning by computer*, tech. rep., Carnegie-Mellon University, Pittsburgh, PA, Dept. of Computer Science, 1968.

[161] W. L. Johnson, H. H. Vilhjálmsson, and S. Marsella, *Serious games for language learning: How much game, how much ai?*, in *AIED*, vol. 125, pp. 306–313, 2005.

[162] R. Oxford and D. Crookall, *Vocabulary learning: A critical analysis of techniques*, *TESL Canada Journal* **7** (1990), no. 2 09–30.

[163] W. Chang and Q. Tan, *Augmented reality system design and scenario study for location-based adaptive mobile learning*, in *Computational Science and Engineering (CSE), 2010 IEEE 13th International Conference on*, pp. 20–27, IEEE, 2010.

[164] J. Orlosky, T. Toyama, D. Sonntag, and K. Kiyokawa, *Using eye-gaze and visualization to augment memory*, in *International Conference on Distributed, Ambient, and Pervasive Interactions*, pp. 282–291, Springer, 2014.

[165] M. Dunleavy, C. Dede, and R. Mitchell, *Affordances and limitations of immersive participatory augmented reality simulations for teaching and learning*, *Journal of science Education and Technology* **18** (2009), no. 1 7–22.

[166] A. Kukulska-Hulme, *Will mobile learning change language learning?*, *ReCALL* **21** (2009), no. 2 157–165.

[167] T.-Y. Liu, T.-H. Tan, and Y.-L. Chu, *2d barcode and augmented reality supported english learning system*, in *Computer and Information Science, 2007. ICIS 2007. 6th IEEE/ACIS International Conference on*, pp. 5–10, IEEE, 2007.

[168] Y. Itoh, J. Orlosky, and L. Swirski, *3D Eye Tracker Source*, . https://github.com/YutaItoh/3D-Eye-Tracker, accessed March 12th, 2018.

[169] R. Budiu, *Multitasking on mobile devices*, *https://www.nngroup.com/articles/multitasking-mobile/, accessed on May 5th* (2020).

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

[171] J. Rekimoto and K. Nagao, *The world through the computer: Computer augmented interaction with real world environments*, in *Proceedings of the 8th annual ACM symposium on User interface and software technology*, pp. 29–36, 1995.

[172] G. D. Abowd, A. K. Dey, P. J. Brown, N. Davies, M. Smith, and P. Steggles, *Towards a better understanding of context and context-awareness*, in *International symposium on handheld and ubiquitous computing*, pp. 304–307, Springer, 1999.

[173] A. Schmidt, M. Beigl, and H.-W. Gellersen, *There is more to context than location*, *Computers & Graphics* **23** (1999), no. 6 893–901.

[174] T. Starner, S. Mann, B. Rhodes, J. Levine, J. Healey, D. Kirsch, R. W. Picard, and A. Pentland, *Augmented reality through wearable computing*, *Presence: Teleoperators & Virtual Environments* **6** (1997), no. 4 386–398.

[175] D. Schmalstieg and G. Hesina, *Distributed applications for collaborative augmented reality*, in *Proceedings IEEE Virtual Reality 2002*, pp. 59–66, IEEE, 2002.

[176] D. Schmalstieg, T. Langlotz, and M. Billinghurst, *Augmented reality 2.0*, in *Virtual realities*, pp. 13–37. Springer, 2011.

[177] C. Harrison, H. Benko, and A. D. Wilson, *Omnitouch: wearable multitouch interaction everywhere*, in *Proceedings of the 24th annual ACM symposium on User interface software and technology*, pp. 441–450, 2011.

[178] R. Hachiuma, C. Pirchheim, D. Schmalstieg, and H. Saito, *Detectfusion: Detecting and segmenting both known and unknown dynamic objects in real-time slam*, *arXiv preprint arXiv:1907.09127* (2019).

[179] K. He, G. Gkioxari, P. Dollár, and R. Girshick, *Mask r-cnn*, in *Proceedings of the IEEE international conference on computer vision*, pp. 2961–2969, 2017.

[180] T. Höllerer, S. Feiner, D. Hallaway, B. Bell, M. Lanzagorta, D. Brown, S. Julier, Y. Baillot, and L. Rosenblum, *User interface management techniques for collaborative mobile augmented reality*, *Computers & Graphics* **25** (2001), no. 5 799–810.

[181] M. Tatzgern, D. Kalkofen, R. Grasset, and D. Schmalstieg, *Hedgehog labeling: View management techniques for external labels in 3d space*, in *2014 IEEE Virtual Reality (VR)*, pp. 27–32, IEEE, 2014.

[182] S. Julier, M. Lanzagorta, Y. Baillot, L. Rosenblum, S. Feiner, T. Höllerer, and S. Sestito, *Information filtering for mobile augmented reality*, in *Proceedings IEEE and ACM International Symposium on Augmented Reality (ISAR 2000)*, pp. 3–11, IEEE, 2000.

[183] M. Tatzgern, V. Orso, D. Kalkofen, G. Jacucci, L. Gamberini, and D. Schmalstieg, *Adaptive information density for augmented reality displays*, in *2016 IEEE Virtual Reality (VR)*, pp. 83–92, IEEE, 2016.

[184] B. MacIntyre, E. D. Mynatt, S. Voida, K. M. Hansen, J. Tullio, and G. M. Corso, *Support for multitasking and background awareness using interactive peripheral displays*, in *Proceedings of the 14th annual ACM symposium on User interface software and technology*, pp. 41–50, 2001.

[185] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, *Ssd: Single shot multibox detector*, in *European conference on computer vision*, pp. 21–37, Springer, 2016.

[186] 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.

[187] 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, (New York, NY, USA), p. 1–12, Association for Computing Machinery, 2020.

[188] R. Godwin-Jones, *Mobile apps for language learning*, *Language Learning & Technology* **15** (2011), no. 2 2–11.

[189] E. Ophir, C. Nass, and A. D. Wagner, *Cognitive control in media multitaskers*, *Proceedings of the National Academy of Sciences* **106** (2009), no. 37 15583–15587.

[190] T. Toyama, D. Sonntag, J. Orlosky, and K. Kiyokawa, *Attention engagement and cognitive state analysis for augmented reality text display functions*, in *Proceedings of the 20th international conference on Intelligent user interfaces*, pp. 322–332, ACM, 2015.

[191] C. M. Privitera and L. W. Stark, *Algorithms for defining visual regions-of-interest: Comparison with eye fixations*, *IEEE Transactions on pattern analysis and machine intelligence* **22** (2000), no. 9 970–982.

[192] M. Ma, H. Fan, and K. M. Kitani, *Going deeper into first-person activity recognition*, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1894–1903, 2016.

[193] Y. Xiong, K. Zhu, D. Lin, and X. Tang, *Recognize complex events from static images by fusing deep channels*, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1600–1609, 2015.

[194] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva, *Learning deep features for scene recognition using places database*, in *Advances in neural information processing systems*, pp. 487–495, 2014.

[195] T. Bagautdinov, A. Alahi, F. Fleuret, P. Fua, and S. Savarese, *Social scene understanding: End-to-end multi-person action localization and collective activity recognition*, in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4315–4324, 2017.

[196] T. Starner, *Human-powered wearable computing*, *IBM systems Journal* **35** (1996), no. 3.4 618–629.

[197] L. Merino, M. Schwarzl, M. Kraus, M. Sedlmair, D. Schmalstieg, and D. Weiskopf, *Evaluating mixed and augmented reality: A systematic literature review (2009-2019)*, in *2020 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pp. 438–451, IEEE, 2020.

[198] S. Irawati, S. Green, M. Billinghurst, A. Duenser, and H. Ko, *" move the couch where?": developing an augmented reality multimodal interface*, in *2006 IEEE/ACM International Symposium on Mixed and Augmented Reality*, pp. 183–186, IEEE, 2006.

[199] H. Hasada, J. Zhang, K. Yamamoto, B. Ryskeldiev, and Y. Ochiai, *Ar cooking: comparing display methods for the instructions of cookwares on ar goggles*, in *International Conference on Human-Computer Interaction*, pp. 127–140, Springer, 2019.

[200] T. Narumi, Y. Ban, T. Kajinami, T. Tanikawa, and M. Hirose, *Augmented perception of satiety: controlling food consumption by changing apparent size of food with*

augmented reality, in *Proceedings of the SIGCHI conference on human factors in computing systems*, pp. 109–118, 2012.

[201] J. O. Álvarez Márquez and J. Ziegler, *In-store augmented reality-enabled product comparison and recommendation*, in *Fourteenth ACM Conference on Recommender Systems*, pp. 180–189, 2020.

[202] P. Knierim, P. W. Woźniak, Y. Abdelrahman, and A. Schmidt, *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*, pp. 1–12, 2019.

[203] T. N. Hoang and B. H. Thomas, *Augmented reality in-situ 3d model menu for outdoors*, in *2008 7th IEEE/ACM International Symposium on Mixed and Augmented Reality*, pp. 185–186, IEEE, 2008.

[204] F. Brudy, *Interactive menus in augmented reality environments*, *Beyond the Desktop* (2013) 1.

[205] Z. Huang, W. Li, and P. Hui, *Ubii: Towards seamless interaction between digital and physical worlds*, in *Proceedings of the 23rd ACM international conference on Multimedia*, pp. 341–350, 2015.

[206] M. Pourmemar and C. Poullis, *Visualizing and interacting with hierarchical menus in immersive augmented reality*, in *The 17th International Conference on Virtual-Reality Continuum and its Applications in Industry*, pp. 1–9, 2019.

[207] D. Lindlbauer, A. M. Feit, and O. Hilliges, *Context-aware online adaptation of mixed reality interfaces*, in *Proceedings of the 32nd annual ACM symposium on user interface software and technology*, pp. 147–160, 2019.

[208] D. Damen, T. Leelasawassuk, O. Haines, A. Calway, and W. W. Mayol-Cuevas, *You-do, i-learn: Discovering task relevant objects and their modes of interaction from multi-user egocentric video.*, in *BMVC*, vol. 2, p. 3, 2014.

[209] J. Orlosky, K. Kiyokawa, T. Toyama, and D. Sonntag, *Halo content: Context-aware viewspace management for non-invasive augmented reality*, in *Proceedings of the 20th International Conference on Intelligent User Interfaces*, pp. 369–373, 2015.

[210] Microsoft, “App views.” Article, March, 2022. Retrieved June 4, 2022 from `https://docs.microsoft.com/en-us/windows/mixed-reality/design/app-views`.

[211] M. Leap, “Landscape design.” Article, May, 2019. Retrieved June 4, 2022 from `https://developer.magicleap.com/en-us/learn/guides/design-landscape`.

[212] K. Lebeck, T. Kohno, and F. Roesner, *Enabling multiple applications to simultaneously augment reality: Challenges and directions*, in *Proceedings of the 20th International Workshop on Mobile Computing Systems and Applications*, pp. 81–86, 2019.

[213] J. Brooke *et. al.*, *Sus-a quick and dirty usability scale*, *Usability evaluation in industry* **189** (1996), no. 194 4–7.

[214] E. Suma, S. Finkelstein, M. Reid, S. Babu, A. Ulinski, and L. F. Hodges, *Evaluation of the cognitive effects of travel technique in complex real and virtual environments*, *IEEE Transactions on Visualization and Computer Graphics* **16** (2009), no. 4 690–702.

[215] J. Cohen, *Statistical power analysis for the behavioral sciences*. Routledge, 2013.

[216] A. Bangor, P. T. Kortum, and J. T. Miller, *An empirical evaluation of the system usability scale*, *Intl. Journal of Human–Computer Interaction* **24** (2008), no. 6 574–594.

[217] B. Bahrami, D. Carmel, V. Walsh, G. Rees, and N. Lavie, *Spatial attention can modulate unconscious orientation processing*, *Perception* **37** (2008), no. 10 1520–1528.

[218] L. R. Harris, M. Jenkin, and D. C. Zikovitz, *Visual and non-visual cues in the perception of linear self motion*, *Experimental brain research* **135** (2000), no. 1 12–21.

[219] S. Julier, M. Lanzagorta, Y. Baillot, L. Rosenblum, S. Feiner, T. Hollerer, and S. Sestito, *Information filtering for mobile augmented reality*, in *Proceedings IEEE and ACM International Symposium on Augmented Reality (ISAR 2000)*, pp. 3–11, 2000.

[220] M. Tatzgern, V. Orso, D. Kalkofen, G. Jacucci, L. Gamberini, and D. Schmalstieg, *Adaptive information density for augmented reality displays*, in *2016 IEEE Virtual Reality (VR)*, pp. 83–92, 2016.

[221] H. N. Gabow and R. E. Tarjan, *Faster scaling algorithms for general graph matching problems*, *Journal of the ACM (JACM)* **38** (1991), no. 4 815–853.

[222] P. Voigt and A. Von dem Bussche, *The eu general data protection regulation (gdpr)*, *A Practical Guide, 1st Ed., Cham: Springer International Publishing* **10** (2017), no. 3152676 10–5555.