Understanding How to Translate from Children's Tangible Learning Apps to Mobile Augmented Reality through Technical Development Research

by

Shubhra Sarker

B.Sc. in Computer Science, Khulna University, 2016

Thesis Submitted in Partial Fulfillment of the
Requirements for the Degree of
Master of Science

in the
School of Interactive Arts and Technology
Faculty of Communication, Arts and Technology

SIMON FRASER UNIVERSITY
Fall 2019

Approval

Name:	**Shubhra Sarker**
Degree:	**Master of Science**
Title:	**Understanding How to Translate from Children's Tangible Learning Apps to Mobile Augmented Reality through Technical Development Research**
Examining Committee:	**Chair: Marek Hatala** Professor
	Alissa Antle Senior Supervisor Professor
	Bernhard Riecke Supervisor Associate Professor
	Steve Dipaola External Examiner Professor
Date Defended/Approved:	December 17, 2019

Abstract

In this thesis, I discuss the design and development of two Augmented Reality (AR) applications derived from two tangible systems. In this technology development research, I explore if it is feasible to port tangible systems to mobile (tablet-based) AR systems so that these systems can be more widely deployed as research prototypes and eventually as products. I use two existing tangible systems (Youtopia, PhonoBlocks), which have been validated empirically, as case studies. To do this, I begin by determining the key requirements that each AR system must have – those known through theoretical design guidance or shown in previous studies to be important for the effectiveness of the tangible system. I designed and implemented design and technical AR solutions for each requirement. For some features, I explore possible solutions and provide a rationale for the selection of a solution. For other features, I present one solution that is feasible. In this way, I explore feature by feature if it is feasible to create AR applications that are more affordable and scalable than the tangible systems while keeping the core design requirements. Future work would need to include the integration of these features and creating fully functional systems. I discuss the technical and design challenges for each of the applications and possible considerations to make when making similar applications. I also contribute preliminary design guidelines for creating new tabletop AR learning applications. Overall, my result contributes to new techniques that may be used to create a tablet-based AR application, which is more affordable and scalable for technology-enabled learning research and development than tangible systems or AR through head-mounted displays.

Keywords: Augmented reality; tangible user interfaces; digital tabletops; collaborative learning; language learning. technical development research.

Acknowledgements

I would like to thank Dr. Alissa Antle for all her guidance, time and mentorship throughout this educational journey. I would also like to thank Dr. Bernhard Riecke for his help and support in making this thesis possible. I would also like to thank everyone from the TECI lab. The opportunity to work at this lab has been the best learning experience of my life.

I would also like to give a special thank you to my parents and my family for all their support and encouragement which allowed me to pursue such an incredible academic opportunity.

Table of Contents

Contents

List of Tables

List of Figures

List of Acronyms

SFU	Simon Fraser University
LAC	Library and Archives Canada
AR	Augmented Reality
VR	Virtual Reality
TUI	Tangible User Interface
TEL	Technology Enabled Learning
CCI	Child Computer Interaction
CV	Computer Vision
HMD	Head Mounted Display
SLAM	Simultaneous localization and mapping

Chapter 1.

Introduction

1.1. Overview

Tangible User Interfaces (TUIs), as a learning tool, have been very popular within the child-computer interaction (CCI) community. However, in many cases, TUIs need custom and/or expensive hardware integrations, which raises the question of the scalability of the system. Implementing a TUI with less hardware dependency could help to overcome this problem. In this research, I will go through the technology development research process I used to explore developing features for two augmented reality (AR) applications based upon requirements taken from two existing tangible systems. The design and technical challenges I faced in this process are those one might face while transforming many types of tangible systems into mobile (tablet-based) AR applications. I will also present the design decisions I made to implement key requirements for the two systems, discussing the problems I faced, and finally then validating my solutions through proof of concepts. I contribute technical approaches and design considerations for researchers and developers porting tangible to mobile AR systems, and for those developing mobile AR systems in general. I also touch on UX decisions I made that may generalize to the design of collaborative AR applications.

1.2. Research Problem

Hands-on interaction and collaborative experiences are often considered beneficial for child education within academic research. Mediating learning environments with technology has become more popular in the last decades due to advancements in science and technology. Tangible technologies like TUIs and digital tabletops have received attention in education because of their many promising features. For example, TUIs enable physical object manipulation that aids spatial problem-solving tasks (Antle, 2013a). A tangible system can be designed to enable enhanced sensory experience (e.g. touch), facilitate embodied interactions (e.g. cognitive offloading, two-handed interaction), and support multiple users (e.g. physical object sharing), all of which may improve learning

(Antle, 2013b). For these types of reasons, researchers have been motivated to create a range of tangible systems for children's learning.

In this thesis, I am going to talk about two of them. The first one is *PhonoBlocks,* a tangible system that uses embedded dynamic colour cues with 3D lowercase tangible letters to help children learn to read and spell six alphabetic rules of English (Antle et al., 2015). A key feature of this system is that tangible letters change colour and sound the moment when adding a new letter to a word, based on rules of the alphabetic principle. PhonoBlocks was tested with eight children (five boys and eight girls) aged 7-8 years old who were at risk for dyslexia (Fan et al., 2017). The students were familiar with English letter names and basic sounds. However, they had minimal knowledge of how to read and spell words using the six rules of the PhonoBlocks system. The results from a case study showed that all children made "significant gains in reading and spelling on trained and untrained(new) words, and students could apply all spelling rules a month later" (Fan et al., 2017). It was also tested with ten Mandarin-speaking children who were learning English, with similar positive evidence (Fan et al., 2018a)

The second system is *Youtopia*, a land-use planning activity that allows two children to learn about sustainability and resource use in a face-to-face environment through the exploration of tangibles and dialogue within a digital world (Antle et al., 2013). Two key features of this system are the interdependency between land-use types and the assignment of roles and tools for creating those land-use types to different users. The Youtopia system was evaluated through a study with twenty pairs of 5th-grade children who used the Youtopia tabletop system in pairs to design a world they would want to live in (Wise et al., 2015, Wise et al., 2017). In order to explore design features related to collaboration in learning, half of the pairs were assigned to a positive interdependence condition in which they were assigned a role (natural resource manager or human developer) and associated tools based on their role. The other half were not assigned roles or tangible tools. The results showed that "pairs in the assigned roles/controls condition gave more in-depth explanations to their partners about what they wanted to do in the game but did not negotiate with each other more frequently than control pairs. They also had fewer but longer instances of jointly resolved conflict" (Wise et al., 2017). Overall all the children met learning outcomes using the Youtopia system, and the positive interdependence design was shown to have benefits for collaborative learning.

Although the results from evaluations of both these tangible systems were promising, the systems are not scalable or affordable in school environments because they required custom hardware (tangibles, digital tabletop), which are expensive and not readily available. It is also not easy for other researchers to use this system for further study, since they are unique and not easily duplicated. This results in a need for systems that are more accessible at a reasonable cost, and that retain key features shown to be effective for learning. Beginning to address this problem is the focus of my thesis research.

1.2.1 Role of Augmented Reality in Supporting Learning

Augmented Reality (AR) has received a lot of attention recently in the educational technology research community. AR promotes several functions, which makes it a promising tool for early learning. One of the unique features is that AR (1) *enables interaction with virtual and real objects and learn by doing*, both of which may increase attention and motivation (Singhal et al., 2012). Researchers have denoted several other advantages of AR for educational applications. AR (2) *provides a sense of reality* (Lin and Wang, 2012) *and provides natural experiences to visualize complex relationships* (Wu et al., 2013). Visualizing complex relationships with a sense of reality may promote better involvement; hence, it may result in better understanding and improves overall learning (Gandolfi et al., 2018). Research has shown that AR (3) *increases students' participation through its fun way of interaction* (Wojciechowski and Cellary, 2013). Other studies have shown that AR learning (4*) provides motivation and facilitates understanding* (Ivanova and Ivanov, 2011), (5) *supports communication and collaborative learning* (Yuen et al., 2011), (6) *increases spatial ability* (Wojciechowski and Cellary,2013) and (7) *enhances problem-solving skills* (Boonbrahm et al., 2015).These potential benefits of AR for learning overlap with those of TUIs.

With the enhancement of computer vision technology and highly configured mobile phone and tablets, handheld mobile AR has taken on a new turn. Because mobile devices are affordable, portable and scalable, handheld AR has been getting more popular in recent years. These motivated me to transform our tangible works into AR mobile applications. The goal of maintaining the core design aspects or features of each TUI system created a significant challenge to implementing equivalent features without custom tangible or expensive physical objects and on the small screen of devices like a smartphone or tablet.

1.2.2 Augmented Reality as a Language Learning Tool

Many researchers have been working on AR applications to promote word learning in a fun and effective way. Spanish researcher Juan and his colleagues introduced Learning words using AR (Juan et al., 2010), a marker-based AR platform to teach children (age 5 to 6 years) Spanish words. This system had a lot of custom hardware (camera, head-mounted display, computer) and marker-based cards. The cards hold flat 2D markers and five different learning games. They evaluated the system by a study with 32 children aged 5-6 years old. The result indicated that the students had fun playing the game and learned to spell words through this approach. Brazilian researchers presented the design and evaluation of ARBlocks (Silva et al., 2013) to help children from the age of 4 to 8 to improve their reading skills of English. They conducted a controlled group study that shows students using AR learn faster and get more engaged than students using traditional lessons. Billinghurst et al. from the University of Washington presented a mixed reality interface called the MagicBook that used a real book to seamlessly help users transport between reality and Virtuality (Billinghurst et al., 2001). This system had a hand-held display computer with one or more physical books. Later on, this concept was used by (Mahadzir and Phung, 2013) to develop an AR pop-up book to teach English storytelling to children in Malaysia. The results from observational and interview data showed that AR technology increased the attention and engagement of the children. Many other researchers on early language learning have also shown the potential of AR as a learning tool (Boonbrahm et al., 2015; Fan et al., 2018b). Based on some of these findings and the unique features of AR, there is a potential to use AR for early language learning.

1.2.3 Augmented Reality as a Collaborative Learning Tool

AR can promote social interaction among physically collocated users by populating virtual objects and enabling natural means of communication (speech, gestures, etc.) (Kaufmann, 2003). Early researchers have found AR as an enhancing collaboration tool between students and instructors and among students (Billinghurst, 2012). Much research has been done to find challenges to design and develop collaborative AR applications. In 2005 Klopfer and his team from MIT designed two different AR games to compare design challenges requiring *positive interdependence, promotive interaction, individual accountability, interpersonal and small group skills, and group processing* (Klopfer et al., 2005). The result shows direct and indirect collaboration among teams as well as players in the same team. More recent studies have also shown the potential promise of AR as a collaborative learning tool. For example, in a co-design study of a primary school AR textbook by Alhumaidan et al. (2018), they showed the positive influence of AR on driving students' attention, increases in their motivation through engagement and collaborative problem solving through a 3D augmented reward schedule. Not a lot of the literature uses multiplayer AR experiences to support a collaborative learning environment, but those who have tried it have mentioned several benefits of using networked collaborative experiences (Ortiz et al., 2018). Many earlier works needed custom hardware or expensive AR headsets to accomplish a networked AR setup. However, with the development of mobile phone hardware and advances in deep learning and computer vision, networked or multiplayer AR has been become an active field of research. This has made technology affordable and better performing than ever before. Based on some of these finding and the unique features of AR, there is a potential to use AR for collaborative learning.

1.3. Thesis Guide

In this thesis, I address the challenge of porting key features of two tangible systems for children's learning to mobile AR. My research questions are, “Is it feasible to port key features of effective tangibles systems for (language, collaborative) learning to equivalent features in mobile AR?” In Chapter 1, I outline this research gap and provide the context for my work. In Chapter 2, I provide my research motivation, provide background about the two tangible systems I used as my case studies in technology research, and analyze background literature related to AR for learning. In Chapter 3, I discuss my methodology, which includes a description of my technical research method and how I have validated my solution. In Chapter 4, I discuss how I have approached each design requirements with corresponding technical challenges and design decisions. In Chapter 5, I discuss the technical and design implications with AR design guidelines for tabletop AR applications for children. In Chapter 6, I conclude the thesis by providing a summary of my research goals and contributions as well as discuss the limitations to my study and the potential for future research in the area of inquiry.

Chapter 2.

Literature Review

2.1 Research Motivation

In the previous section, I introduced the two tangible systems that my collogues from the TECI lab developed. While both of those tangible systems had some promising results in terms of learning, it is hard to scale the projects for classrooms and home environments. My goal was to determine if I could adapt the key design features from these projects to mobile AR platforms, which might enable the further development of more affordable and scalable deployments.

To make the system **affordable,** I need to consider the following. (1) *The use of less or no custom hardware* will make the system easily affordable for school or home environments. For instance, a user can buy letter blocks from the dollar stores and use the app in their mobile phone/ tablet to get the system working. (2) *The application should be compatible with all AR supported phones and tablets* to make it easily accessible to a variety of users. (3) *The application should be easy to use from both technical and interaction perspective* as our target users are likely not technology experts.

To make the system **scalable,** I need to consider the following. (1) *The system should use widely used tools, plugins and software development kits (SDKs)* so that it is readily available for all current and future compatible devices. This will also help the system to be easily upgraded if there are any future updates on those tools, plugins and SDKs. (2) *The system should use state-of-the-art, handheld AR technologies* to make sure they don’t quickly become obsolete in the near future.

2.2 Framing the Problem Space

I derived the requirements for this problem space from existing literature on the two projects. There are three categories of requirements for each system. Meeting these requirements required specific design decisions and interface features that enable specific kinds of interactions. In the AR versions of each system, the design decisions and features may be different. However, each system should address, as much as possible, the original requirements. First, each system had specific learning goals and associated ***learning***

design requirements related to 1) learning to read/spell (PhonoBlocks), and 2) supporting collaborative learning about land-use planning (Youtopia). Learning goals require specific design features to enable the kinds of interactions that support learning. While the design features may be different for the new AR systems, the learning goal-based requirements and kinds of interactions the interface and system should enable will be similar. Second, each system also follows the ***best interaction design practices*** for (1) hands-on learning with tangibles (Antle, 2007) and (2) collaborative learning with tangible tabletops (Antle et al., 2011) These are non-specific to the domain of learning, and some are non-specific to tangibles since they result in general types of beneficial interactions. Third, each system contains ***specific features*****,** which should be replicated to ensure that the resulting AR system is commensurate with the TUI systems, which were shown to be effective, and so they can, for example, be used as a research instruments in comparative studies as the first step in the validation of the system on a new platform.

2.2.1 Case 1: PhonoBlocks Tangible Reading

PhonoBlocks was primarily developed based on four best practices of multisensory instruction derived from the phonological deficit theory (Ramus, 2003). The primary goal of the system was to facilitate reading and spelling English words for children of age 7 to 8 at risk for dyslexia. The system was comprised of six letter-sound rules, which are critical for early reading and spelling, and all students must master these alphabetic principles (i.e. rules). In Fan et al, they state, “The design of PhonoBlocks, in particular, its two core design features, dynamic embedded colour cues and 3D tangible letters, were developed based on theories of causes of dyslexia and analysis of noncomputational multi-sensory reading interventions, which are effective but resource-intensive.” (Fan et al., 2017). They posed the following research questions related to investigating the effectiveness of PhonoBlocks: “RQ1: Do children improve word reading and spelling accuracy after instruction with PhonoBlocks on trained words, new words, and on both after a month?” They are wanted to understand the contribution of key design features to any successful outcomes. In a second question, they asked, “RQ2: What are the key design factors in PhonoBlocks that benefit children in learning to read and spell?”. To further understand how and why the system may support learning, they also asked,“RQ3: What do children like and dislike about the system?” and “RQ4: Do children’s individual characteristics influence learning performance, behaviours, and/or likes/dislikes?” (Fan et al., 2017).

Fan et al. describe PhonoBlocks as follows: “PhonoBlocks is comprised of a touch-based laptop display, a word-making platform with seven slots, and 46 lowercase “hand-sized” 3D tangible letters (duplicates for common letters, e.g. a, e, d, t). Children learn letter-sound correspondences by placing one or more 3D tangible letters on the platform. Visual feedback is embedded in the 3D letters using LED strips that change colour to indicate sound changes as letters are added or removed (e.g. Figure 2.1). Visual and audio feedback is also provided on the digital display using coloured 2D letters and playing associated letter sounds.” (Fan et al., 2017).

Figure 2.1 One of the lessons (Magic-e) from PhonoBlocks showing the change of colour to indicate the sound change.

2.2.2 Requirements for an AR PhonoBlocks

AR PhonoBlocks (ARPB) will address as many of the requirements from the tangible PhonoBlocks system as possible. Tangible PhonoBlocks had a learning goal of supporting children to learn to read and spell words using six rules of the alphabetic principle. The focus was on investigating if tangibility could enable beneficial learning processes related to **hands-on interaction with tangibles**, and as a result, improve learning outcomes. This resulted in four core **learning design requirements** (Fan et al., 2016). (1) *The system should focus on letter-sound correspondence within short word contexts*. This led to several features. The input to the reading application is through an interface that includes 46 lowercase “hand-sized” 3D tangible letters and a seven-letter word making slots on a platform, in a line, one slot for each letter. PhonoBlocks also provided blending (i.e. audio pronouncing the whole word) and decoding (i.e. audio of each phoneme sound) functionality on the touch screen with individual and blended letter colours as well as sounds. (2) *There should be support for hands-on multimodal interaction*. The 46 lowercase “hand-sized” 3D tangible letters and a word making platform promote letter tracing and two-handed physical manipulation of letters (e.g. organizational strategies), as well as ease of interaction (e.g. placing letters). That is important to note here that the physicality of the letters was important to preserve while transforming PhonoBlocks from the tangible to AR. Previous studies show that hands-on manipulation and tactile interaction with physical objects can enhance learners ability to remember things for longer periods of time (Treiman, 2017) and can also help to engage learner's attention (Marshall, 2007; Fan et al 2017). So while designing the AR version, it was important to preserve the physical letter shapes used in the system rather than making the whole system digital. For example, if the letter shapes had been 3D CGI models rather than physical objects, then the benefits to learning of hands-on manipulation and tactile interaction would have been lost. (3) *There should be a way to draw attention to the letter-sound correspondences within words*. This led to the feature of creating dynamic, transparent letters that change colour the moment placing one letter changes the sound of another in the word. (4) *There should be support for tutor - learner interaction.* PhonoBlocks contains both tutor and student modes allowing the tutor to teach each level to the child and enabling the child to practice on their own.

PhonoBlocks was designed following from some **best practices** for tangible learning design (e.g. (Antle and Wise, 2013)), which should also be considered in the AR PhonoBlocks requirements. Since the system is meant to be used by more than one user (e.g. teacher, student), (5) *there should be multiple access points*. That is, there must be ways to encourage interaction and make it easy to do so for two people. This led to the large size of the input space and the offline working area (e.g. could organize letters on the table before inserting into slots). The system should use (6) *physical constraints* to encourage or enforce correct input (e.g. correct orientation of the letters p,q,b,d) as long as this didn't detract from learning. Note that letter reversal is not specific to children with dyslexia but is common to all children, most of whom learn the correct orientation without interventions. This led to the use of knobs in the letter's slots and magnets to the letter would easily fit into a slot only in the correct letter orientation (e.g. d/p, q/b). The system (7) *should leverage the use of multi-modal representations of letters*. This led to the use of letter-sound representations as sounds, pictures, 3D physical letter shapes and 2D digital letter symbols. The system should leverage the (8) *spatial properties of tangibility*, either through direct, isomorphic or metaphoric relations between physical representations and semantic meanings. This led to the use of the letter slot interfaces that ensured that children place letters in a specific linear order, with a space for the first letter, second letter etc.

Another feature **specific** to PhonoBlocks was that the letters were (9) *small cases only*. This was a requirement taken from interviews with dyslexia tutors who were involved in system design, but this need not hold for all tangible reading systems. It is specific to our problem space.

The following table summarizes the design requirements for tangible PhonoBlocks.

	Design Requirements for PhonoBlocks	
1.	Focus on letter-sound correspondence	The system should have letter-sound correspondence within short word context.
2.	Multiple interaction modality	Letter tracing and physical manipulation; ease of interaction.
3.	Draw attention to letter-sound correspondences	Dynamic colour and sound changes.
4.	Teacher-student interaction	A teacher can teach through tutor mode; Children can practice through student mode.
5.	Multiple access points	Encourage interaction for two people; large input space size.
6.	Physical constraints for correct letter orientation	Knobs in the letter slots and magnets to ensure the correct letter orientation.
7.	Multiple letter representations	Letter-sound representations as sounds, pictures, 3D physical letter shapes and 2D digital letter symbols.
8.	Spatiality (linear)	Letter slot interface to ensure the linear ordering of letters.
9.	Use of small case letter	Small case letters as a requirement provided by the dyslexia tutors.

Table 2.1 Design requirements for tangible PhonoBlocks.

2.2.3 Case 2: Youtopia Tangible Tabletop System

Youtopia is a collaborative learning activity that was developed to promote learning about sustainability. The goal of the system was to "meet basic BC (Canada) learning outcomes for grade 5 environment and sustainability topics (ages 10-11)"(Wise et al., 2015)" . The application enables explores research questions focused on the design of collaborative learning on a tabletop and multitouch tangible tabletop interaction. Youtopia is the third extension of the original work, called Towards Utopia, a single-player tangible tabletop sustainable land use planning activity developed in TECI Lab, SIAT, SFU (Antle et al., 2011). The second prototype was called Futura, a collaborative multi-touch tabletop sustainable land use planning activity implemented on a custom digital tabletop (Antle et al., 2011a). A study was run at Winter Olympics to find issues in collaborative game-based learning. Youtopia was developed to address issues of these previous works and come up with some design implementations that support effective collaboration. The issue was that simultaneous multiple-user interaction alone could not ensure collaboration (Dillenbourg, 1999, Kreijns et al., 2003). The study conducted for this Youtopia system had the following research questions: "Does assigning children interdependent roles/ tools in Youtopia lead to increases in **RQ1:** working together? **RQ2:** talking in-depth about the sustainability domain? **RQ3:** resolving conflicts jointly rather than unilaterally?" Youtopia tries to solve this by creating an opportunity for positive interdependence through codependent access points.

Antle et al., (2013) have stated the activity of the Youtopia system as following. "The main activity in Youtopia is using physical stamps to designate land use types on an interactive map …. The goal of the activity is to support either a small or large population with enough shelter, food and energy without over polluting the world. There are different types of shelter, food and energy sources, as well as nature reserves, each with different benefits and limitations. The map is of a small area of land, including mountains, valleys, grasslands and a river. The game begins by default with a small population and default map. There are four maps that have similar size and resources. Only the terrain elements are arranged differently. Choosing a new map by touching the maps symbol on the menu restarts the game. Choosing a large population by touching the population symbol on the menu continues the same game with a larger population or restarts depending on which is selected. Together, the different populations and maps add sufficient complexity to the application that children can play for long sessions (Wise et al., 2015)"

The results from an experimental study of Youtopia (Wise et al., 2015) showed a significant improvement in student's understanding of the key concepts at $p<0.001$ level and that students collaborated effectively, including engaging in in-depth discussion of land-use values and bi-lateral resolution of conflicts (Wise et al., 2017).

Figure 2.2 (a) Stamping trees into lumber. (b) Groups of a related tree and wrench stamps. (c) World state and food circle touched.

2.2.4 Requirements for an AR Youtopia System

The Youtopia tangible tabletop system had a learning goal of supporting pairs of children (aged 10-11, grade 5) to learn about land use planning collaboratively. The focus was on investigating if **tangibility** could be used to support **positive interdependence,** which would enable rich dialogue between including negotiation and conflict resolution related to their land-use planning task. This resulted in three core **collaborative learning design requirements:** (1) *The system should have codependent access points*. That is, more than one input is required to achieve system response (e.g. stamp forest to lumber first, then stamp lumber into houses). (2) *The system should have two sets of input objects that can be assigned to children in roles* (human developer, nature resource manager) *or remain unassigned* and available for either child to use.

Tangible Youtopia was designed following from some **best practices** in collaborative and tangible learning design (Antle and Wise, 2013), which should also be considered in the AR Youtopia requirements. To provide support for pauses for reflection, (3) *the system should be designed to have features where interaction is paused, and children are provided with a reason to reflect*. For example, for the tangible Youtopia reflective pauses were supported through the following design features: (a) information tool: freezes interaction and provides information on a specific resource and (b) impact tool: freezes interaction and provides information about current world state in terms of pollution and population’s need for food, house and energy, and “pig” asks, “Is this a world you want to live in?” To avoid a head-down interaction, which can lead to parallel activity: (4) *the system should be designed to encourage children to monitor each other’s activity,* and (5) *the system should be designed to encourage children to monitor each other’s gaze.* To support the development of shared understanding, (6) *the system should have a referential anchor* that is context-specific representations that support children to come to a shared understanding of the problem and solutions. For example, for the tangible Youtopia, the dominant referential anchor is the interactive map. And lastly, to support collaborative activity, (7) *the system should have objects of negotiation*, which are external representations that can be modified by the pair or individuals during the learning process. For tangible Youtopia, these are the input stamps for natural resources and human developments.

Another feature **specific** to Youtopia was (8) *the number – 13 -- of input objects* (i.e. land-use stamps), and that (9) *these stamps could only be used by one child at a time*. (roles condition) However, this is specific to our problem space, and the number of stamps could be more or less for other applications.

The following table summarizes the design requirements for Youtopia Tabletop.

Design Requirements for Youtopia Tabletop		
1.	Co-dependent access points	More than one input required for system response (e.g. stamp forest to lumber first, then stamp lumber into houses).
2.	Interdependent assigned roles/tools	Two sets of input objects are assigned in alignment with particular roles (human developer, natural resources manager) or remain unassigned for either child to use.
3.	Pause interaction with reason to reflect	(a) Information Tool: Freeze interaction and provides information on a specific resource. (b) Impact Tool: Freeze interaction and provides information about the current world state.
4.	Activity monitoring of each other	Avoid head down interaction to prevent parallel activity; monitor each other’s activity.

5.	Gaze monitoring of each other	Avoid head down interaction to prevent parallel activity; monitor each other's gaze.
6.	Referential anchors	Context-specific representation for a shared understanding of the problem and solutions. (e.g.Interactive maps).
7.	Objects of negotiation	External representations can be modified by the pair or individuals during the learning process. (e.g. Input stamps for natural resource and human developments.)
8.	Number of input objects	There should be 13 input objects (i.e. land-use stamps) in the system.
9.	Use of stamp one child at a time	Stamps could be used by only one child at a time.

Table 2.2 Design requirements for Youtopia tangible tabletop.

2.3 Introduction to AR

The term "**Augmented Reality**" (AR) was considered to be first coined by Tom Caudell and David Mizell as a training tool for Air Force pilots in 1990 (Berryman, 2012). However, the concept of AR could be found before 1990. In fact, in World II, the British Military used the concept of AR to display radar information on the windshield of a fighter plane (Berryman, 2012; Vaughan-Nichols, 2009). In 1966, Ivan Sutherland introduced the first head-mounted display that merged computational information with reality. In 1968, Sutherland and his lab created, The Sword of Damocles, which is considered as the first AR system (Berryman, 2012). AR gained a lot of interest in the 1970s and 1980s, which resulted in new innovations and increased researcher involvement in this technology in the 1990s. Roland Azuma defined AR in 1997 as a combination of virtual and real objects coexisting in same space (Azuma, 1997). Because the definition was too narrow, Azuma updated his definition identifying three properties of AR: (1) it combines virtual and real objects in a real environment; (2) it can distinguish virtual and real objects with each other; (3) it runs interactively in real-time. In other words, AR provides information not present in the real world by adding virtual objects into real scenes (Sayed et al., 2011). Chen and Tsai (2012) supported this definition and framed AR as interaction with 2D or 3D virtual objects in real-world environments. On the other hand, Wojciechowski and Cellar (2013) defined AR as an extension of VR (Virtual reality) where users can interact with virtual objects in the real world rather than full immersion inside a virtual world.

With the improvements of modern smartphone cameras and the availability of hardware technologies, AR research more quickly began to emerge, starting around 2007 (Carmigniani et al., 2011). In 2009, ARToolkit (Kato and Billinghurst, 1999) was created and started its journey with basic fiducial marker tracking and browser supports. Between the years 2013 to 2017, there have been some remarkable developments in the field of AR. In 2014 google introduced Google Glass, the very first commercial AR glasses. The technology was not as successful as hoped for. In the same year, Google introduced Project Tango, a high-end AR computing platform for mobile devices like tablets and smartphones. However, Google stopped support for Tango in December 2017 in order to focus on the mass market. In March 2018, Google introduced ARCore, which would not need custom sensors for AR experiences, unlike Tango. In the meanwhile, Apple had also introduced ARKit in June 2017. With all this development in AR technologies, AR has become more prominent than ever before. For example, modern AR head-mounted displays (HMDs), like Microsoft HoloLens and Magic Leap, are getting more popular in the research and user community. However, because of the very high price of the HMDs and the rapidly improving high definition available through new smartphone cameras, smartphone AR is now getting more popularity day by day. Moreover, with improvement in computer vision algorithms like Simultaneous Localization and Mapping (SLAM) technology and Visual Inertial Odometry, mobile AR has become more accessible and more stable than ever before.

2.3.1. Technology Enhanced Learning and AR

Technology Enhanced Learning (TEL) has become popular in the last decade. TEL combines emergent technologies like ubiquitous learning, edutainment, games, AR and learning analytics with enriched multimodal learning environments (Johnson et al., 2016). In particular, AR has gained attention in recent years in TEL research (Masmuzidin and Aziz, 2018; Wu et al., 2013). This has resulted in a variety of TEL publications related to AR in different learning domains. Some proposed benefits for AR are that AR enhances attention and motivation (O'Brien and Toms, 2005; Ivanova and Ivanov, 2011) by enabling rich interaction (Azuma, 2004) and providing natural or real-world experience to the user. Besides, it makes learning fun and offers experiences that are impossible to achieve in real life (Wojciechowski et al., 2004). Not only that, research has shown that learning with AR may increase spatial ability (Cheng and Tsai, 2013), improve problem-solving skills and promote collaboration (Billinghurst, 2012; Yuen et al., 2011). I took these promising benefits of AR into account when working to convert our previous two TUI systems into new AR systems.

2.3.2. Scalable and Affordable Technology

One of the reasons behind the increasing interest in AR is the development of affordable and scalable software and hardware that supports AR technology. Almost all modern Android phone and tablets support AR technology. This helps researchers and developers reach more users with mobile AR applications, systems and tools. Mobile AR uses off-the-shelf technology that contains less or no customized hardware. Smartphones and tablets are becoming ubiquitous, and many schools already use them in classroom instruction for children (personal conversation, Antle 2018). Apps for these devices can be designed to support AR using just the camera feature. This makes functionalities of AR applications more easily scalable than fully tangible systems. This is possible because AR applications use Computer Vision (CV) as a foundation, and it’s easy to add new features to the application by implementing a particular CV algorithm. Besides, CV technologies are getting “smarter” with the involvement of deep learning. This has resulted in new SDKs and plugins that have made mobile AR development both easier and more stable than ever before.

2.4 Related Works

In this section, I discuss prior works done in AR for learning related to the two tangible cases. I have divided the section into two parts. In the first section, I will discuss some of the previous work on AR as a language learning tool. In the second section, I will focus on related works on collaborative learning and AR tools. In both sections, I will highlight key design features and discuss the findings or results of the work, noting any limitations.

2.4.1. Related Works on AR Language Learning

Many AR and VR reading applications have been developed for children. However, very few have focused on teaching them the alphabetic rules of English. In this section, I will discuss the prior work and their limitations and to set the stage for how my solution is unique from the existing ones.

Spanish researcher Juan and his colleagues introduced Learning words using AR (Juan et al., 2010), a marker-based AR platform to teach children (age 5 to 6 years) Spanish words. The system used a camera, a head-mounted display, a computer, and a set of squared markers. The children had to put the markers under the camera and look through the head-mounted display to see the augmented scene. The tutor could see the same scene through a computer display. The study was conducted with 32 children (age 5-6 years) who played the AR game and equivalent real game. This system shows a potentiality of AR as a learning tool for students. It also provided a simple interaction and effective design aspects. For example, it used virtual cartoon characters to give feedback to the students, which motivated children to learn. The use of both head mount display (for user/children) and computer display (for the tutor) helped the tutor to monitor the learning session of the children and help them if necessary. In fact, the tutor monitoring is one of the design requirements for AR PhonoBlocks as well. However, the shortcoming of this work is (1) using a lot of hardware devices (camera, head-mounted display, computer) (2) using a marker-based solution (set of square card markers). In my proposed system, I am going to (1) minimize the hardware usage (only a tablet) and (2) make the system compatible with many varieties of markerless physical letter sets.

Brazilian researchers presented the design and evaluation of ARBlocks (Silva et al., 2013) to help children from the age of 4 to 8 to improve their reading skills of English. This was also a marker-based solution where a webcam detected the markers on the block, and a projector showed the augmented reality contents. The experimental study showed some promising results for students who used AR over the traditional method. While this system was a good indication of the possibility of an AR system, it did not provide any good design ideas for an AR reading system. For example, the block design does not enable letter tracing or provide shape cues because letters were printed on blocks.

Billinghurst et al. from the University of Washington presented the MagicBook, a mixed reality interface that used a real book to seamlessly transport users between reality and Virtuality (Billinghurst et al., 2001). The system had a hand-held AR display, a computer and one or more physical books. The system was not designed for children, but it was a solid demonstration for designing an augmented book. Later on, this concept was used by (Mahadzir and Phung, 2013) to develop an AR pop-up book to teach English storytelling to children in Malaysia. The results from observational and interview data showed that AR technology increased the attention and engagement of the children. Though my system is not focusing on storytelling, these papers provide some evidence for the effectiveness of AR systems related to children's reading and language learning.

Mobile Augmented Reality technology in assisting English learning for primary school students was introduced by (Boonbrahm et al., 2015). the paper gave some insight into the technical solution. Though it used marker-based AR to detect the letters and words, it gave me some base ideas on developing spelling games in AR, for example, how to display the 3D animations. However, this study only used capital letters, which have less ambiguity than small case letters, which can be easily confused (a and o; c and e, etc.). The game does not support letter-sound correspondence. Finally, the maximum length of a word a child could make is four letters. So, the main design characteristics that make my work unique from this are (1) markerless AR with small case letters. (2) letter-sound correspondence and (3) use a maximum of six-character length words.

From the prior works I have discussed above, it is noticeable that none of them used a markerless AR solution for their system. While this can be a good choice to make the system stable and less complex from technical aspects, it makes the system-dependent on custom hardware. This is one of the main scopes that I will work in my study. Secondly, the designs provided by most of the systems left gaps in fulfilling learning objectives. My colleagues designed PhonoBlocks based on theories dyslexia, best practices of multisensory instruction, and research on the specific advantages afforded by TUIs for hands-on learning (Fan, Antle, and Cramer 2016). The two key design features of PhonoBlocks are (1) embedded dynamic colour cues and (2) 3D tangible letters. In my solution, I am going to replace the tangible letters that used custom hardware, with any physical letter sets available in the schools or commercially available in the market.

2.4.2. Related Works on Collaborative AR Applications

AR technology has been shown to have potential benefits in supporting a range of collaborative learning situations. Many researchers have studied how AR can enhance collaborative learning in different ways. This includes collaborative AR games, collaborative design etc. However, only a few of the works have discussed the affordances of their AR system for supporting collaboration. In addition, many of the works use custom or expensive hardware. In this section, I will discuss prior works and their limitations and how my solution is unique from the existing ones.

One of the earliest work on collaborative AR learning was done by (Klopfer et al., 2005) where they talk about different design challenges to promote collaborative learning. Two different AR games were used for the study where they focused on design challenges (or games) requiring: *positive interdependence, promotive interaction, individual accountability, interpersonal and small group skills, and group processing.* The first game named Environmental Detectives (ED) was a Real-World Location-based AR role-playing game (RPG) game where the students would play as environmental scientists. They would work on teams of two to three players, identify contaminants, chart their path through the environment, and devise possible plans for remediation if necessary. The game was designed in a way that a single player could not obtain the required data in the allocated time. So, they had to work together as a team, gather data and find a solution together (positive interdependence). Each of the teams had a Pocket PC, a walkie-talkie, a printed map and a notepad. Individual members of a team were assigned particular tasks with the tools which promoted strong collaboration among team members. However, there was not much collaboration between two teams as both of them were using the same application with total control over the system. This system very complex and has a lot of learning components. However, it demonstrates the effectiveness of splitting tasks up to ensure positive interdependence.

Another early research project by Regenbrecht et al.(2002) showed some interesting ideas about user interaction with an AR headset in a workspace for a meeting. In the paper, the authors discussed interaction techniques using tangible desktop items in a collaborative tabletop setting. The author used a system called “Magic Meeting” (Regenbrecht et al., 2002), which uses an HMD AR display, which can detect different markers. The output of each marker is reflected in the HMDs and two external displays. In this paper, the authors specifically discuss the collaborative interaction techniques used in this system. Although this work is slightly old and uses technology that is now obsolete or improved to a large extent, we are interested in this work for its design decisions. First of all, the use of a head-mounted display opens up the chance to use both hands for tangible interactions. A unique interaction technique mentioned in the paper is “Cake Platter,” which is simply a turnable, plate-shaped device that is used as the central location for shared 3D objects. Users can turn the plate for a different view of the augmented object placed on top of it. Another interesting interaction technique is the 2D-3D linkage. This means the player can select a 3D object from the 2D window or get information about the 3D model in a 2D window. The author mentions about four users using the system; however, no validation of the proposed interaction techniques for collaboration was found in this paper.

To promote greater collaboration between groups, two new games were developed (Charles River City (CRC), which combines environmental science and epidemiology to create a largescale investigation, and Mad City Murder (MCM), which uses the ED premise to create a mystery investigation). Some new features were introduced in these games, including Distinct Roles. Each team's information is explicitly described to them as only a small piece of the puzzle, and they need information from other roles to solve the problem. This sharing is facilitated by the infrared beaming of information. Distinct Roles facilitated the positive interdependence between teams.

Alhumaidan et al. (2018) presented the design and evaluation of an AR textbook for collaborative learning experience through a co-design process technique called “co-operative inquiry” (Druin, 1999). The study involved nine primary school children (five females, four males aged 8 to10) and three adult participants of different academic backgrounds. The project had multiple phases, including low-tech prototyping, co-design process, actual AR app session and formative evaluation. We are particularly interested in design decisions. For example, markers were placed separately at the edge of the textbook and could only be seen if two books were joined together. The augmented content that was shown by tracking the marker was necessary to complete that lesson. As a result, the students had to join each other to complete the AR marker in order to be tracked by the camera and display the AR scene (Figure 2.3). This feature supports collaboration with positive interdependence. Another interesting feature, which resulted from the children's idea, was "AR reward cards." Students would get a reward card at the end of one textbook activity. This card plays the animation of the corresponding object drawn in the card. Students tend to join and help each other to complete a lesson and get a reward card. Moreover, students could join two reward cards by placing it side by side to display enhanced AR content.

Figure 2.3 Joint AR marker for positive interdependence. Copyright - Alhumaidan et al. (2018)

The process of validating the design of this system included a cooperative inquiry and layered elaboration study. The process started with the children trying out the AR textbook. The children were asked to write down their likes, dislikes and suggested designed ideas after a short exploration. Three groups were formed, and each group was facilitated by an adult member. The adult members were asking open-ended questions to the students to elaborate on their ideas and writing the notes. The second session was based on a layered elaboration study to collect data focusing on a collaborative experiment. In this session, each group came with some ideas on how they could improve the design. The design suggestions were noted by the adult members of each group. All the transcripts from the groups were collected and analyzed (thematic analysis) by the adult members with the presence of the researchers. The dataset was categorized in like, dislike and suggested design ideas. The adult members then discussed the interpretations of each transcript to agree on ideas based on collaborative experience, learning and usability. The result of the analysis provided insight for future design features and guidelines. The authors had identified some key design features from the formative evaluation of the co-design process. While this showed some helpful guidelines that can be implemented in school textbooks for a collaborative learning experience, it had some limitations. For example, the study mainly focused on a single tablet device shared by multiple students, which the authors suggested was chaotic at times. Although the authors mention using multiple tablets for more than one student, it is not clear if how this would work or if the tablets would be networked or not.

Shin et al. (2018) presented a collaborative living room design application called Share Design that runs on multiple synchronized and spatially aware tablets for couples. The goal of their work was to investigate the use of AR for user designing together. To achieve this, the researchers used the Google Tango tablets (i.e. tablets enabling the Tango platform developed by Google, primarily for AR computing platform) to design and develop the application. Six couples participated in the study, where they designed an office space together that had the size of a typical Korean living room. The study was designed into three sections, (1) exploring ideas individually, (2) explaining ideas to each other, (3) designing together. The authors analyzed their design process through the role couples took and their way of communication through observational data and a post-interview session. The design, prototype and the study suggested some useful implications for an application that deals with the collaborative application. One of the important design features is "multiple workspaces." This facilitated collaboration by utilizing a single coordinate

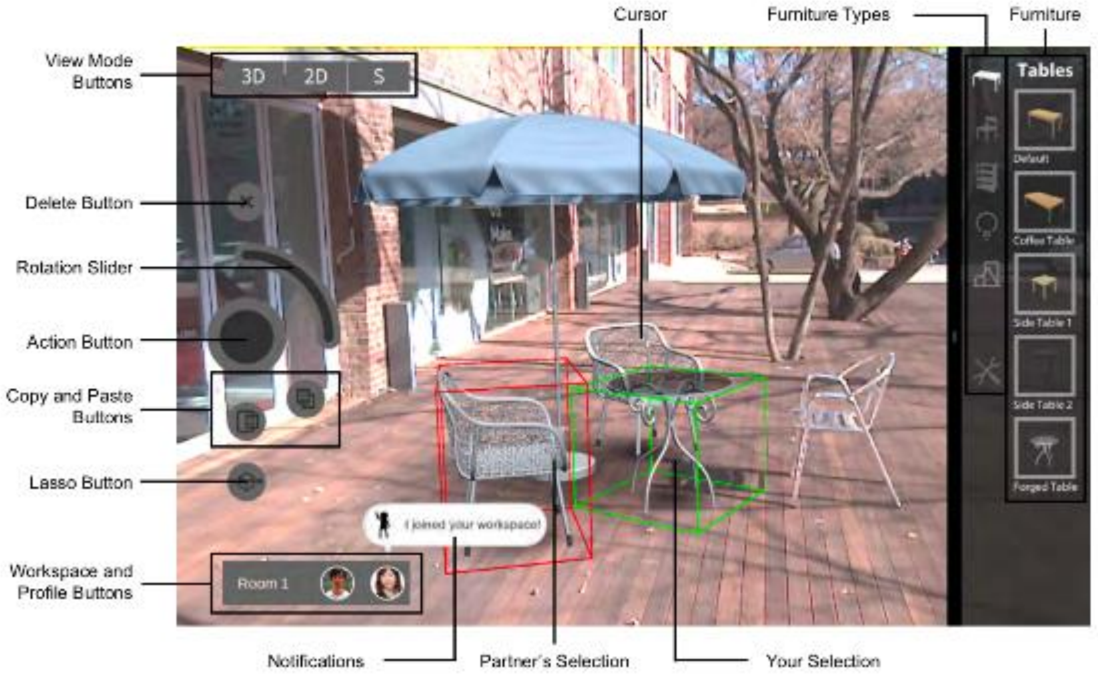


Figure 2.4 The UI of Share Design, showing a collapsible side menu on the right-hand side and a collaboration menu on the left- hand bottom edge. Two virtual objects are placed in the real world and selected, with the local user's selections highlighted. Copyright - Shin et al. (2018)

system shared between tablets. Couples could invite each other to join in the same workspace and merge their ideas. Also, the objects placed by each user were differentiated with colour. The authors used on-screen UI for selecting, rotating and spawning objects on the world space (Figure 2.4).

Another interesting feature is the spatial understanding, which has three modes. (1) AR Mode: Shows AR contents in the world space while moving the tablet; (2) Plan View Mode: Shows a top-down map of the room; and (3) Couch Mode: Reviewing the virtual snapshots taken while designing. The interesting part about this feature is that the modes changed with respect to the tablet tilting angle. Holding table horizontally (Plan View Mode) reduced arm fatigue (Ahlström et al., 1992) when using the AR Mode for an extended period of time. As the system was spatially aware, any space that is used by one can not be used by another user. This constraint leads to the dependency between the users when one needs to change a layout. The results indeed showed that there was a lot of discussion between the couple while designing layouts for different areas. It also implied that there were reflective pauses when one person talked or modified the layout. The post-interview showed some positive feedback from the users, specifically when it came to user interaction. Also, the authors mentioned, from their observational data, that there were reflective pauses from time to time, although the users were using two different tablets.

One thing to notice from the technical point of view is the use of Google Tango devices to develop “Share Design.” Tango devices were spatially aware, meaning the tablets had the ability to map the surrounding environments with respect to its position and orientation. Tango could do it through its depth sensor, and AR-enabled hardware design. This makes networked AR a lot easier. However, Google has stopped producing Tango from 2017, and we plan to develop our application for commonly used and available mobile phones and tablets (iOS and Android). This makes our solution space challenging; however, it is important for us to develop the app in commonly available devices to promote the affordability and scalability of our system.

2.5. Handheld AR and Usability for Children

While handheld (mobile) AR has been shown to have potential benefits to engage early learners in an affordable and scalable way; however, some researchers have shown usability issues of this technology for young children. Some of the core issues with AR for child interaction are predicted and explainable by child development psychology areas such as (1) *motor skills* (as most of the interaction is in 3D space); (2) *spatial cognition* (as the user should understand the spatial relationship between physical and digital object); (3) *attention control* (user should have the ability to distinguish between augmented and nonaugmented object interaction); and (4) *logical thinking and conceptualization* (user should have the ability to understand the augmented contents are computer generated and not real.) (Radu et al., 2016). Other usability issues a child might face are mostly related to physical interaction with the handheld AR device. The physical constraints include the following: (5) *Inability to hold the handheld AR device properly* (Hornecker and Dünser, 2008). AR uses computer vision techniques to collect important information from the environment (called feature point) and thus enable tracking. Continuous shaking of the handheld device can hamper the feature tracking process. So, holding the device steady is important for handheld AR. However, handheld AR devices, particularly AR tablets, can be large for a child's hand to hold. This issue leads to a child often shaking the device, resulting in bad tracking, hence, bad AR experience. Another related issue is (6) *Lack of precise interaction with augmented objects through the 2D screen space (Hornecker and Dünser, 2008)*. Currently, there are two different common ways of creating and manipulating augmented objects in the handheld AR platform: (a) Finger interaction: User can touch on the screen, which creates objects at that point. To manipulate a particular object, the user needs to move the camera to that object and simply touch on it; and(b) Crosshair interaction: A fixed crosshair is generated in the mobile screen, which indicates the selection point. Users can then tap on a side on-screen button or anywhere on the screen to trigger a selection. Both of the methods have usability issues in terms of precision (I will talk about this in detail in my design rationale section). However, research has shown that finger interaction is faster than crosshair-based interaction (Radu et al., 2016). Another issue is: (7) *Inability to remember the spatial position of the augmented objects along with other things*. This problem happens because the AR content is invisible as the user moves her camera. In a room-scale AR (AR application, which is location independent), objects are generally augmented in a 3D spatial trackable area. With more objects spawned in the environment, it is sometimes easy for children to remember where they have put particular objects. Researchers have also found that age is a big factor related to children’s AR experience, with older children having a better experience. In a study with 5-10 years old children conducted by Radu et al. (2016), the researchers showed that younger children make slower selection times, more tracking losses and longer time to recover lost trackings. A further issue is (8) *Excessive and/or poor content quality* can cause a perceptual issue (Santos et al.2015). An excessive amount of content can distract the user from the core interaction loop and make the perception of key information difficult. Clarity of the content is also important to facilitate users attending to and perceiving the relevant information for learning (Santos et al.2015).

Chapter 3.Technical Development Research Methodology

In this chapter, I am going to discuss the methodology I am using for the validation of my systems. I will start by summarizing the methodological approach I am using and then discuss different parts of the validation method. Finally, I will conclude with a description of the approach and methods in the context of my research.

3.1. Technical Development Research

For my thesis, I am taking a technical development research methodology or approach. Technical development in HCI research is a process of creating a technical prototype and showing that it works as required through one or more proof of concepts. In HCI research, the focus is more on interface and interaction components more than system design (e.g. algorithm design). In this type of technical research, the work is considered to be done when it is validated through a working implementation. This method can be considered in three parts (Hudson and Mankoff, 2014): (a) concept creation; (b) validation through proof-of-concept implementation; and, if applicable and within scope; (c) secondary validation. A working technical solution should have the fully completed output of (a) and (b). While (c) can be helpful for further validation of the creation, it is not always necessary for technical research methodology. In summary, technical development research can be defined as the following – “Technical research method is the process of creating something that implies knowledge of something new that works, as well as reusable knowledge to create a similar class of creation or even multiple class of different things.”(Hudson and Mankoff, 2014).

3.1.1. Research versus Development

Technical research sometimes is misinterpreted as development; however, they are not the same. Both technical research and development result in the output of technology creation. However, the goal or purpose behind the creation differs between research and development (Hudson and Mankoff, 2014). Technical research provides knowledge gained by creating something new. The knowledge should be reusable for similar applications. That is, the knowledge should have the quality to contribute to the creation of similar or multiple artifact classes. For example, the creation process of a markerless AR version of PhonoBlocks can provide knowledge for developing similar OCR (Optical Character Recognition) applications for other language alphabets. On the other hand, development is the creation of some tool or product using previously created knowledge. In addition, the knowledge gained in creating the tool is not necessarily reusable in other contexts. Both my cases involve the creation of new knowledge through my technical solutions, which may be generalized to other similar AR applications.

3.1.2. Inventive Research

Technical research relies on invention, which is the process of creating and bringing new things into the world. The process of the invention can be done by either combining things that are already known to create larger and complex things that did not exist in the past or coming up with something completely new, which was not created before (Hudson and Mankoff, 2014). Inventive HCI research should focus on the creation that either meets end-users' needs or enables a creation possible, easy or cost-effective. Different forms of contributions can be made through inventive research. The overall two categories of contributions are (a) direct creation, and (b) enabling creation based on (Hudson and Mankoff, 2014).

a. **Direct Creation:** Direct creation is the type of inventive research where the creation both meets human needs and provides new knowledge for future inventions. Contributing to direct creation should promote a solution that should directly meet an end user's goal. In most cases, direct creation involves the creation, including something that:
   - improves aspects of long-standing goal (e.g. supporting collaborative work at a distance (Engelbart and William, 1968); (Ishii et al., 1994));
   - introduces new capabilities (e.g. interacting with wall displays that are larger than the reach of a person's arms ((Shoemaker et al., 2007)); or
   - brings a capability to a new user population (e.g. photography by the blind ((Jayant et al., 2011)).

b. **Enabling Creation:** Enabling creation is another form of inventive research method where the "creation" does not necessarily aim for end-user need, rather enables others to address a need to make it possible, easier or less expensive. This form of research can come in different forms (Hudson and Mankoff, 2014):

- **Tools**: Tools enables the easy creation of a certain class of things. Tools do not meet end-users need rather makes it easy and quick for other developers to achieve the goal of an end-user.
- **System**: System combines a set of capabilities into a single working whole. Systems generally come up with abstraction to make these capabilities more useful, manageable and easier to deploy/use.
- **Basic Capabilities:** Advance on a specific and difficult problem that is holding up progress in a problem domain. This form of research includes the creation of new algorithms or design that can solve a problem in a particular problem domain.
- **Import and Adapt:** The import and adaption technique simple imports advances made in other technical areas and putting them to use for new purposes. While there can be some argument about this process being an invention or not, it surely must be considered as a technical research advance. One example of this method can be seen in the introduction of finite-state automata for HCI by Newman, (1968), which was originally devised to model neuronal activity (McCulloch and Pitts, 1943). So, importing and adapting powerful techniques can be valuable and should be considered as a contribution of its own.

For my thesis, I am using the enabling creation method, where I have imported and adapted currently available technology to create affordable AR learning applications. To be more specific, I have adapted the technology behind CNN based OCR, ARCore cloud anchor enabled multiplayer and image segmentation algorithm for colour augmentation. All these technologies are highly used in many different applications, such as self-driving cars (image segmentation), object detection and classification (CNN) etc. However, I have imported these ideas and applied them to the research area of technology-enabled learning. I have discussed more the methodology used for both of my applications later in section 3.3.

3.2. The Process of Technical Research

The process of technical research can be broken down into ***three*** parts: (A) Concept creation; (B) Proof-of-concept implementation; and (C) Secondary validation (Hudson and Mankoff, 2014)

3.2.A.Concept Creation: The first part of the process of technical research is for the researcher to come up with an idea of what the creation is going to be. This phase is important as the implementation process can be highly affected by this process. There are two different approaches to concept creation.

- **Needs First:** In a need first method, the researcher should start from an observed human need and find a technical approach to make a positive impact on that need. For this method, it is important to think about the feasibility of the concept. This means, whether or not the idea is doable with currently available technology and how that might help the user to achieve their goal.
- **Technology First:** In this method, the researcher should specialize in a particular technology and find what human need can be fulfilled through that technology.

For my thesis, I am using the Needs First method. I have gathered general (e.g. scalable, cost-effective) and specific requirements related to end-user interactional features for two systems that need to be converted from one technology platform to another more accessible one. I have tracked down the problems inherent in each conversion, and then for each of the problems, I have found a solution(s) that meets the application-specific and general requirements (i.e. is scalable and affordable). I have discussed more the methodology used for both of my applications later in section 3.3.

3.2.B.Validation through Proof-of-Concept Implementation: The primary validation method for technical research is the implementation of the concept as a proof-of-concept (i.e. a working prototype of a feature or set of features or system). This technique can be really powerful based on the quality of the proof-of-concept (Hudson and Mankoff, 2014). Hudson and Mankoff (2014) have emphasized this method as following, "The centrality of proof-of-concept implementations as a validation mechanism is so strong that the evolved value system gives *building* a central role.

Even a really strong user study or other empirical evaluation cannot improve a mediocre concept (or tell us how good an invention it is). In contrast, a proof-of-concept implementation is a critical form of validation because an invented concept is not normally trusted to be more than mediocre without an implementation."

However, to be a sufficient validation method, the proof-of-concept needs to be complete enough to answer, "does it work (well enough)" and address a set of additional questions we might ask in a secondary (extended) evaluation. How we define "well enough" depends on the "type and extent of the implementation we undertake" (Hudson and Mankoff, 2014). One way of defining "well enough" is to find evidence indicating that the creation promotes advantages over existing solutions for the same problem space. When we have the workable proof-of-concept, it is then important to find out in what circumstances does it work. This is important for the proof-of-concept to be robust to widely varying conditions in the real world; that is, the creation/research should enable the creation of a wide range of other things to be created. This means the knowledge combined on proof of concept should generalize similar development approaches. However, if a "system" does not work well, we can "still learn something useful if there is enough promise that the concept might be made to work and we uncover information about what problems need to be overcome" (Hudson and Mankoff, 2014)

3.2.1. Type of Proof-of-Concept Implementation

There are some cases where the proof-of-concept method might not be suitable. For example, if the technology being developed is a concept that is way ahead of its time, that is that no enabling technology exists, then a proof-of-concept cannot be used to validate the idea or concepts (see next section). However, when a concept is well coupled up with available technology, proof-of-concept implementation can be a valid approach. There are several methods to create a proof-of-concept of a concept with existing enabling technologies. The methods are categorized by their amount of robustness. These are some of the methods widely used in proof-of-concept implementations sorted by the least to most robust:(a) description in prose; (b) presentation with images; (c) video showing the invention in use; (d) live demonstration by the inventors; (e) testing of properties with users; and (f) deployment to others to use independently. The deployment method is the most powerful and robust method to create a proof-of-concept validation. However, it is important to examine and find the tradeoffs between robustness and completeness compared to the cost and effort necessary to create a “fully functional” implementation. Fully functional implementation of the proof-of-concept clearly provides evidence about the quality of invention. Besides, a high level of completeness is almost the same as the final product. However, the problem with this approach is that it needs a lot of effort, possibly on many aspects of the system unrelated to the research invention. On the other hand, presentations with image and video showing the invention’s use are both times effective and also gives the user an overall idea of the system closely. Considering these, I choose the method (b) and (c) as my primary validation method demoing different features of my applications. Both of the applications, however, has been open-sourced [1] for deployment as a part of future work.

When we have a system that already works, and we want to extend it (e.g. for new technology), one idea to highlight the innovative pieces of the creation rather than creating a complete application. We can derive the key requirements for features that ensure that the older system is effective works and implement each of the features using a different technology. This implementation provides knowledge about it is viable to create a previous system with different or newer technology. This approach is time-efficient and takes less effort to validate a system than creating the full system on the new platform. It provides specific knowledge based on our previously workable system, so the increment of additional knowledge is prominent with this approach. Besides, when we focus on specific requirements, each of these pieces becomes robust and complete. My proofs-of-concept were developed based on some pre-defined requirements from the older workable system and has been deployed through open-source for others to use independently. I have not pursued feedback on the deployment, saving this for future work.

In summary, I chose to focus on creating, testing functionality and then making available (deploying through open-source) several proof-of-concept implementations of key functionality for each of the two cases. Each proof-of-concept was created to meet the requirements for key features that are needed for usability, and/or were shown in user studies to be essential for system effectiveness. The task of porting the entire system for PhonoBlocks and Youtopia is beyond the scope of my thesis. Instead, I chose to focus on creating several proofs-of-concept requiring innovation to determine if the approach to transform from tangible to hybrid physical-AR would be feasible for these two kinds of

---

[1] https://github.com/Shubhra22/TUI-to-AR

features that may be shared by many tangible learning systems. I have created one complete prototype for AR PhonoBlocks that satisfies design requirements discussed in Chapter 2. More about the prototype and individual design implementations are discussed in Chapter4. For the AR Yoputopia, I have created 3 different prototypes, each exploring different design requirements for the system. Some of the design decisions have been changed over time with new versions of the prototypes. However, some of the design requirements for AR Youtopia has remained unsolved. More about the prototypes and individual design decisions are discussed in Chapter 4.

3.2.2. Alternative Proof-of Concept Implementations

There are situations where the proof-of-concept method may not technically work (Hudson and Mankoff, 2014). For example, situations where the concepts require an enabling technology that does not yet exist. Creating such an application can be time-consuming, costly and sometimes almost impossible to do. However, because the idea is ahead of its time does not means the idea is not valid. This kind of situation can be approached with the following methods.

a. **Buying a time machine:** This method can be done by spending a large sum of money to access state-of-the-art technology, which will be affordable in the future. For example, many AR headset devices (e.g. Magic Leap[1], HoloLens[2]) are now too costly and gives better AR experience than mobile devices. Now coming up with some idea that is hard to do in mobile devices can be replaced by buying a costly headset and assume the price will be affordable in the future.
b. **Wizard of Oz:** This technique involves simulating advanced capabilities through a hidden human who performs an action that a future system might be able to provide autonomously.
c. **Simulation:** Simulating some or all of the actions related to the system through a number of human workers. This method has emerged in the form of crowdsourcing.

For my thesis, I have not used any of the alternative proof of concepts discussed here. The reason behind it is that I have used all the currently available technology to make the applications affordable for the end users. Also, all the technical problems for the apps are approachable with available tools and techniques.

3.2.C.Secondary (Extended) Validation

While the primary validation in technical research gives the answer to,” Does it work?” a secondary validation method can provide answers to how good or bad it works with respect to particular scenarios. There are several methods to approach a secondary validation of the system. Some of them are problem-specific, and some are more generic. In this section, I am going to talk about the commonly used secondary validation technique in the technical research method.

a. **Usability Test:** The first technique is called the usability test. This is a widely used technique for validation. This method is mostly used for systems that are considered as “Direct Creation,” that is, the creation directly affects the end-users ' needs and goals. Usability tests offer relatively less assistance in the act of inventing/ conceptualizing a new thing rather offer how well an invention work for end-users. While this technique is really popular in the research community and has its own promises, “the ability of some invention to be modified, extended or applied to a different purpose may be more important than usability” (Olson and Kellogg, 2014).
b. **Human Performance Tests:** Human performance tests refer to the measuring performance of typical users on some set of tasks. This kind of method is a good way to test the system with a narrow and well-defined task. In a controlled environment, the results appear to be mostly valid. This leads away from the wide applicability of the results, which contradicts the usefulness of the invention (e.g. an invention useful for a wide range of tasks).
c. **Machine Performance Tests:** This kind of method is applicable to test an algorithm or a system’s performance. The tests are generated through a simulation of different tasks and conditions. There are different kinds of tests like load testing, stress testing, soak testing, spike testing etc.
d. **Expert Judgement:** This method can be used by taking a review from experts on a particular field about the invention made. This method can be time efficient and gives the researcher an overall insight into the project. However, this method depends on the subjective opinion of experts and, as such, is not

very reliable and repeatable for technical research (Hudson and Mankoff, 2014).

There are some other secondary validations that might apply for indirect creations. Because indirect creations do not necessarily deliver end users' goals, these methods are mostly focused on system performance and developer's usability. Some of the methods are: (a) threshold, ceiling, and breadth of coverage: this includes the creation of examples of invented tool and explains how the tool can be easy to use than existing ones, how the tools are making the creation process easier etc.; (b) presenting a good abstraction: this includes representing the system with some example test uses (a typical validation for good abstraction is done through a set of illustrative examples); (c) usability for developers: this basically is done through usability test on the developed toolkit with developers of that field. This method is similar to the usability test for end-users, but in this case, the end-user is replaced by a developer.

3.3. Method used for PhonoBlocks AR and Youtopia AR

For both PhonoBlocks AR and Youtopia AR, the process of my technical research is done through two steps: (a) concept creation and (b) proof-of-concept validation. I have not done any secondary validation for the system. The method was chosen based on the different aspects of technical research validation discussed above. My systems are the extension of two existing systems (tangible PhonoBlocks and tangible Youtopia) that is already tested and validated through primary and secondary validation methods. My goal for this thesis is to validate the technical and design implications of this two-existing system. I don't intend to validate the usability or the effectiveness of the system. That is why I have used only the primary validation technique for my these through the deployment of proof-of-concept method. The knowledge from the creation process of the system provides a generalizable knowledge for the development of similar applications.

(a) **Concept creation:** The ideas for both of the projects are generated though Need First Method. From previous studies with the tangible version of both of these tools, we have seen the promises of the system. At the same time, there has been a prominent need for these inventions to be affordable and scalable. We took that into account and came up with the idea of the handheld AR application for both PhonoBlocks and Youtopia. The key design features discussed in Chapter 2 are the concepts that I have used to create the proofs-of-concept. Some of the key requirements are implemented in the AR version, and some are not depending on the technical and design challenges I faced for each of them. I will discuss each of the requirements with their challenges and solutions in the next chapter, where I describe my system in detail.

(b) **Proof-of-concept validation:** For the proof-of-concept validation, I choose the image and video presentation method to show different parts of the system working. I have implemented different design ideas (based on requirements discussed in Chapter 2) in the Youtopia AR project, which provides knowledge for creating AR applications that aim to support similar collaborative forms of interaction. For AR PhonoBlocks, I have done multiple critique sessions with my lab mates and supervisor through a live demo of the one complete prototype that I have created. Based on the feedback from the presentations, I have developed the newer version of the prototype. The AR Youtopia was divided into three different proofs-of-concept each focusing on different design requirements of the application. The validation process was similar to AR Youtopia. I presented each design requirement separately through different prototypes and improved design based on the feedback. All the prototypes (one for AR PhonoBlocks and three for AR Youtopia) are also open-source in GitHub with the code for other developers to use it. Currently, the deployed open-source code is not validated through other developers' feedback. This is going to be one of my future works.

(c) **Secondary validation:** For this thesis, I have not conducted any secondary validation. The purpose and scope of my thesis was to find out whether or not it was technically feasible to port key features of existing tangible system into equivalent AR applications. The prototypes that I created can be used to validate technical development but at this very early phase the prototypes are not yet ready to be tested on actual users. So rather than finding out “how good it works for users”, the goal for my thesis was to find out “does it work”, which is the very first stage of technical development and the scope of my thesis. Another reason for not doing for a secondary validation with child users is due to the ethics of working with vulnerable children. Since the systems are at an early technical development stage, usability testing or other evaluation with vulnerable children (specifically for children with dyslexia) is not ethical, hence not a good idea. That is why, I did not do the secondary validation at this time; keeping it as a part of the future work when the systems have been shown to be feasible and complete.

Chapter 4. System Design and Rationale

In this section, I am going to discuss and provide the rationale for the system architecture and design features that enabled the required functionality and interactions for the cases of the AR PhonoBlocks and Youtopia systems. I will start with a description of the system, which will include the architecture of the system, technical implementation and core interaction processes. Then I will discuss different possible design solutions, the trade-offs and decisions I considered while developing the system to meet the requirements (see Chapter 2). Finally, I will discuss the final, particular designs I have chosen for individual system features and the rationale behind those design choices.

4.1. Case 1: AR PhonoBlocks - System Architecture

The PhonoBlocks AR system was developed by focusing on implementing the core design functions and interactions to meet the requirements of the tangible PhonoBlocks system. This system was created using Unity3D[2] for the core development, OpenCV[3] for image segmentation and Tensorflow[4] for developing the CNN model. All the graphics and code used for this project is open source and can be found in Github[5]. The code can be used for any personal and commercial purpose maintaining the GNU General Public License 3.0[6]. The AR system consists of two mandatory components, meaning they are both required to make the system run: (1) an AR mobile application, and (2) a set of 27 x 3D lower-case physical letters. There is also a non-compulsory component (meaning this is not necessary to make the system run, but using it can enhance the performance of the system), which is a tablet stand. The core interactional processes include manipulation of the letter blocks by the user, scanning letters/words once placed by the user using the tablet camera as system input and providing an augmented overlay on top of the letters as AR feedback based on system rules. I will discuss more on each process with corresponding design considerations in the next section. In this section, I am going to

---

[2] https://unity.com/

[3] https://opencv.org/

[4] https://www.tensorflow.org/

[5] https://github.com/Shubhra22/TUI-to-AR/tree/master/ARPhonoBlock

[6] https://choosealicense.com/licenses/gpl-3.0/

discuss the core interactional processes and system architecture and describe technical implementation details.

The first part of the system is the AR mobile application. The device requirement was selected based on the availability of the technology, so it is available for large segment of the population. For example, the user would need an Android/iOS-enabled tablet or mobile device with at least Android 7.0 (Android Marshmallow) or iOS 9.0 version to run the application. The specs were chosen so that the application is supported in most of the current or three to four-year-old Android or iOS devices promoting affordability. Similar to the tangible PhonoBlocks application, the AR application has two modes: (1) teacher mode and (2) student mode. For each mode, there are six different lessons. When a user starts a new lesson, the device camera is activated. The device camera captures the video stream, capturing an image every 20 frames and sends the byte data to a CNN (Convolutional Neural Network) model for text detection. Some image processing (see below) takes place prior to sending the image to the CNN model to make the detection process faster. The output text from the CNN model is then sent to the colour cue algorithm adapted from tangible PhonoBlocks. Here the system changes the colour of each letter based on its location in the word. I will discuss more on the colour cues in the next section. Finally, the letters are mapped with their corresponding 3D model, and an augmented

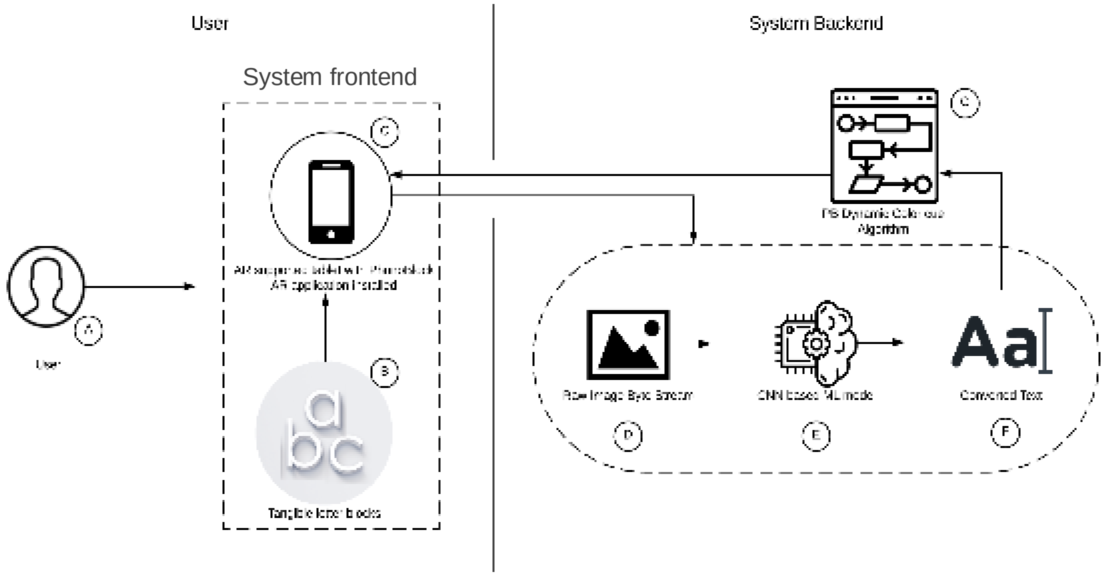


Figure 4.1 System architecture for PhonoBlocks AR.

digital layer is displayed on the top of the physical letter. The following figure (see Figure 4.1) demonstrates the overall system architecture of the PhonoBlocks AR application.

Below I have sequentially discussed the data flow of the system marked as A-F.

A. Users can interact with the front end of the system and have no access to the system's back end where all the computation happens. Users are recommended to use a stand to avoid unintended shaking of the device. A stable device enhances the tracking capability of the system, resulting in better tracking performance.
B. An Android (version 7.0 or higher) or iOS (version 10 or higher) tablet or mobile device can be used to run the PhonoBlocks AR application. The application will use the back-facing camera to detect tangible letters. Users should manipulate the physical letters inside the device's camera frame to make the detection work. All the detection happens at runtime. Similar to tangible PhonoBlocks, the AR PhonoBlocks application has two different modes, each having multiple lessons. Opening a lesson starts the device camera and starts scanning for letters inside the camera's field of view.
C. A camera stream is captured every 20 frames and sent for preprocessing. The preprocessing steps are necessary to remove noise from the image and make it easier for the Convolutional Neural Network (CNN) to detect the letter. The preprocessing is done in the following steps: (1) *thresholding:* the captured images are in RGB format by default. They are first converted into grayscale and then binarized using Otsu's thresholding algorithm(Otsu, 1979) ; (2) *image*

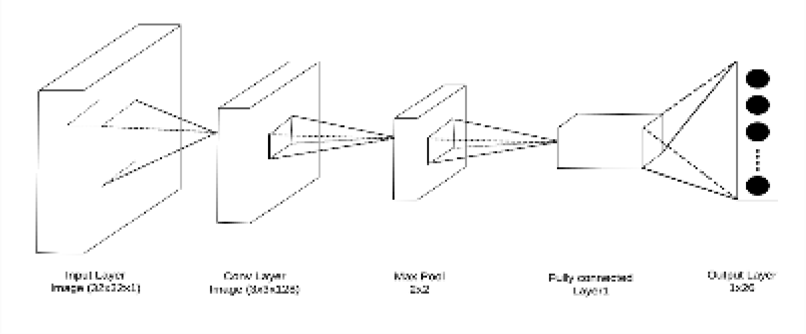


Figure 4.2 CNN model for the character recognition.

denoising: Then, to remove noise from the image, a Gaussian filter was used. Because I am assuming the system will be used in a non-textured single-colour background/tabletop, I have not considered using salt-paper noise deduction; (3) *contour detection and bounding box*: The final step is to find single letters using contour detection and obtain a bounding box containing the main character. To obtain the contour, the image was first passed through the canny edge detector and then a contour approximation method. I used CHAIN_APPROX _SIMPLE method to avoid unwanted computation during the process. The image is then resized to 32x32 (as CNN performs better in lower resolution, the ML model was trained with this image size) and send it to the CNN model (Figure 4.2). The CNN model uses the Chars74k dataset[7], which includes a collection of 74000 characters that are sampled from natural images, hand-drawn characters and computer fonts. Based on the requirements, I used lower-case letters as the training data set. The model itself has one convolution layer, one max pool layers and one fully connected layer. The input layer is binarized, so it has only a single channel layer. The output layer provides a SoftMax layer of the probability of 26 letters. The closest match gets higher confidence; hence, I pass that letter as my final output.

D. The system receives the detected letter from the output of the deep CNN model and pushes the letter to a dynamic list of detected letters. If a letter is already detected, which is determined with its relative position in the real-world and the list, the letter is not pushed to the detected list. From this step, the letter array is passed to the colour cue algorithm, which assigns colours to each letter based on some rules.

E. The colour cue algorithm takes an array of detected letters and applies colour to each of the letters based on some predefined rules. The rules are described in detail in section 4.1.1(R2). This algorithm was first developed by Antle, Fan, & Cramer (2015) during the development of tangible PhonoBlocks. An updated version of the algorithm was proposed by Fan, Antle, & Sarker, (2018) during the development of marker-based AR PhonoBlocks. For my thesis, I have adapted the algorithm from marker-based AR PhonoBlocks as that is the most

---

[7] http://www.ee.surrey.ac.uk/CVSSP/demos/chars74k/

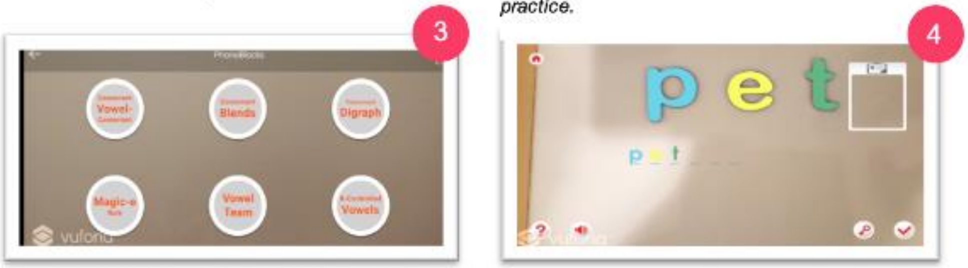
recent work and has already been partially tested on the AR environment. After the colour cue algorithm assigns colours to individual letters, the colour information is sent back to each contour and draws a corresponding colour. These contours are the same ones that were calculated during image preprocessing. After drawing the contours, the texture is passed to a material with an unlit shader, which turns all the black pixels to transparent. This texture is then imposed on the device screen, enabling the colour augmentation process in each letter.

4.1.1. Use Case Scenario for AR PhonoBlocks

In this section, I am going to discuss the use case scenario for AR PhonoBlocks. This includes how the core interaction flow works for a user. To make the process easier

The application starts with a login panel asking user name. For a new user he/she should put a "" in the end of the name for registration.*

Then the user can go either teacher or student mode. Teacher mode is performed in presence of a tutor to teach, where the student mode is for practice.



There are six lessons to choose from, each of the lesson teach particular rules with appropriate color cues.

User can put letter blocks in the camera's view port and the camera would track the changes in runtime. The colors of each letter are changed on runtime.

After making a word, the user can click on the check button to see an animated feedback. The word is also stored in the history bar on top right corner of the screen.

In student mode, the user is given a list of particular predefined words to make. For each of the lessons, user can get up to three hints. User can click on hint button to get a hint.

Figure 4.3 User case scenario for AR PhonoBlocks, assuming the users put the right letters in the right order.

to understand, I have added screenshots of each interaction with a short description. In the scenario, I am assuming that the right letter is placed in the right order. Figure 4.3 briefly discusses a use case scenario of the AR PhonoBlocks System.

4.1.2. Key Design Features and Rationale

The AR PhonoBlocks system has met all the design requirements (R1 to R9) gathered from the tangible system. Because my goal was to make the system affordable and scalable, some of the technical elements were replaced with other design components, which I will discuss in detail later in this section. For example, the custom tangible letter platform was replaced with a set of letters that can be bought from any dollar store. Also, the touch-based computer was replaced with affordable mobile devices (e.g. tablet or mobile phone). However, the core design requirements were kept unchanged with similar functionality. In this section, I will discuss the design features for AR PhonoBlocks, what techniques I tried and which final design I choose and why.

R1. Focus on Letter-Sound correspondence. The system maintains letter-sound correspondence similar to the tangible system. The letters are colour -coded based on their sound. The colour codes are derived from the or the AR version, the customized platform slot has been discarded, and the tangible letter blocks are replaced with easily accessible physical letter blocks available

Learning Activities	Examples
Consonant Vowel Consonant (CVC): CVC patterns	bet
Consonant Blends: two consonants make a blended sound in which you can hear two parts to the sound	f →fl → flag
Consonant Digraph: two consonants make one sound	t (one-time green flash and then off) →th → thin
Magic-e Rule: vowel sound changes from short to long when an *e* is added at the end of word	gam (three-time yellow flashes) → game (three-time red flashes, and then e is off)
Vowel Team: two vowels make one sound	e (one-time orange flash and off)→ea → eat
***Practice Mode**	flag: blag. b is white flash and off

Table 4.1 Six rule-based activities and colour-coding schemas.

in dollar stores. The blending and decoding features are present in the application at runtime when the user keeps a letter in the camera's field of view. All the blending rules are adapted from the marker-based AR version of PhonoBlocks designed by Fan, Antle, & Sarker (2018). Table 4.1 shows the colour cue rules used in the application. The colour cue algorithm uses these rules to make the blending possible. The blending is reflected both on the screen and on the top of the physical letter. When a word is formed, the user can click on the decode sound button located in the lower-left corner to replay the letter-sound correspondence inside that word.

R2. Hands-on Multimodal Interaction. The system uses 46 lowercase 3D letters with a considerable size which satisfies two things: (a) The letters are "hand-sized", that is can easily be manipulated by children, and (b) small enough to fit 6 letters in camera's field of view (FOV) yet big enough for the children to manipulate the letters easily (Figure 4.4a). Holding the tablet for a long time can cause arm fatigue for the children (Munsinger and Quarles, 2019).This can also result in the device shaking and loss of tracking. Besides holding a tablet with one hand and placing letters with the other hand, it can be challenging for children. So, it is recommended to use a stand so that both hands are free while using the application to enhance letter tracing and physical manipulation of the letters. This also promotes ease of interaction with the letters (Figure 4.4b).

Figure 4.4 (a) The letter size was chosen so that at least 6 letters can be fit in the tablet's FOV. (b) Using a tablet stand helps the use of both hands to manipulate letters.

R3. Drawing attention to letter-sound correspondence. While implementing this feature, I have tried multiple solutions. The first approach was simply changing the colour of single letters on the screen as a 2D letter (Figure 4.5a).

The colour changes and blending would happen in runtime, but the change won't be reflected in the physical letter. Although this might work partially, this solution does not completely satisfy the tangible PhonoBlocks design, where the colour change would also reflect directly on the physical letter. To overcome this, I used 3D colour augmentation on top of the physical letters. However, technically achieving this had its own problems. The colour augmentation is done by colouring the black pixels from my binarized captured image (through contour analysis as described in system architecture) and overlay that on top of the letter. However, the colour augmentation won't persist if the character is moved (i.e. I have to continuously colour/redraw the pixels to make it persist, which will cause lag). So, to make the computation faster, I used an unlit shader to draw the contour output calculated during image preprocessing. The shader enables GPU based rendering, which results in faster redrawing. Also, rather than drawing the contours on each frame, the system redraws contours every 20 frames. Although this technique causes a little delay, the delay is barely noticeable, and the colour augmentation is reasonably accurate (Figure 4.5b). To be more specific, the system takes about 0.04 seconds to detect a letter and a frame (0.3 sec) to overlay the colour on the screen (Calculated by putting a timer inside code). However, currently, this delay causes displacement of the overlaying colour when a letter is moved in runtime. Changing colour at runtime promotes drawing children's attention to the moment adding a letter changes the colour of other letters.

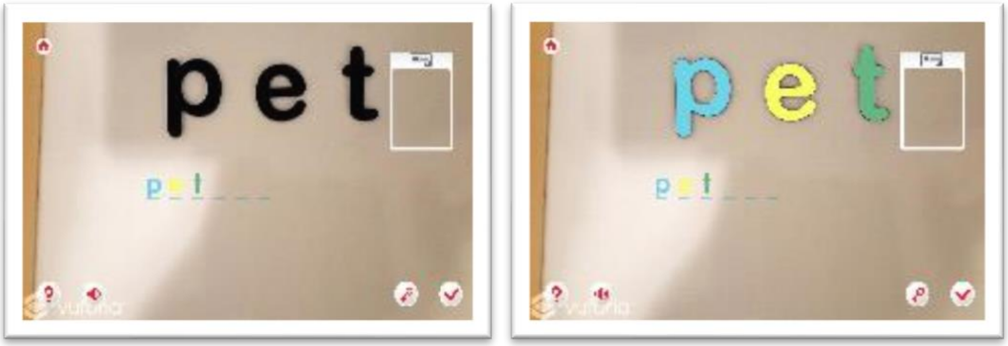


Figure 4.5 (a) Colour coding reflection only in screen space - less attention driven. (b) Colour coding reflection both in screen and world space - more attention driven.

R4. Tutor-learner interaction. Similar to the tangible PhonoBlocks, the AR PhonoBlocks application provides two different modes. The teacher-led model facilitates the tutors to teach each level to the child. The student-led model

Figure 4.6 (a) User can choose between teacher and student led mode.(b) User gets feedback with 3D AR models after completion of each lessons.

promotes self-learning opportunities for the students (Figure 4.6a). Both of the modes have six lessons. However, the student-led model contains three missions for each of the lessons. These missions ask the students to make particular words, provide hints if they are stuck and give feedback at the end of each mission. On the other hand, the teacher-led mode is completely open-ended. It does not have any missions, so the teacher can closely interact with the student through the system without getting interrupted by any system led message. The system does not provide any feedback if the student puts the letters in the wrong order or orientation assuming the tutor will correct the student. However, If the student puts the letters in the wrong sequence in the student-mode, the system plays sound feedback to let them know that they are doing it wrong with hints. The core interaction for these modes was transformed from click-based (on tangible PhonoBlocks) to touch-based (on AR PhonoBlocks). Also, at the end of each lesson, the feedback images were replaced with an augmented 3D model (Figure 4.6b). Showing an augmented 3D model can attract user's attention (Santos et al.2015), promoting more engagement to the system.

R5. Multiple access points. AR PhonoBlocks can be accessed from a different input space as the system does not use any external slots or platform for the letter placement. The system only needs the letters to be placed in the camera's field of view (FOV) with a recommended white background. Thus, the system can be used on any tabletop (Figure 4.7a) or even on a magnetic whiteboard (Figure 4.7b). This enables a more open-ended and accessible input space for a classroom environment.

Figure 4.7 (a) The application can be used on a table top with or without a tablet stand. (b) The application can be used as collaborative classroom work in whiteboard.

R6. Physical constraints. In the tangible PhonoBlocks system, there is an important feature that supports users to place a letter in its correct orientation in each slot. This feature was obtained by custom designing the letter blocks so that similar letters (e.g. d/b, q/p) have a notch that only fits into the base in the correct orientation. However, as I am not using any custom hardware (e.g. base or custom letters), it is hard to provide physical constraints to ensure correct letter orientation for AR PhonoBlocks. In AR PhonoBlocks, this is solved by outputting the same letter orientation as the user's input. For example, if the user uses the letter "p" as "d," the system will automatically consider it as "d" (Figure 4.8). The system assumes that in the teacher-led mode, the teacher will correct the student for any mistake. However, in the student-led mode, if the student put a wrong oriented letter (e.g. p/d, q/b), the system indicates it as a mistake by a black colour overlay as the system can

Figure 4.8 The CNN model can auto detect difference between similar words based on their orientation.

Figure 4.10 (a) Multimodal presentation: letter sound correspondence. (b) Multimodal presentation: 3D colour blending, augmented object and 2D letter symbol.

only detect letters which are put in a correct orientation. One limitation of this approach is that the students won't be able to know if the right letter was placed in a wrong orientation.

R7. Multimodal representation of letters. The system leverages the use of multimodal representation of the letters through sounds (letter-sound correspondence)(Figure 4.9a), 3D augmented models (representations of each lesson's word with 3D model), 3D colour augmentation (augmentation of each letter on top of the physical piece) and 2D digital letter symbol (Figure 4.9b)

R8. Spatiality. This is another design requirement that was achieved using the tangible user interface, which is not present in AR PhonoBlocks. In the tangible PhonoBlocks, the slot interface ensured the linear ordering of letters. Because of the absence of any external hardware, the AR system does not have a way to ensure letters are placed in linear order. However, the letter order can be encouraged by drawing six slots or boxes in a paper or in the whiteboard (Figure 4.10).

Figure 4.9 Remaking slot interface by drawing boxes in the white board.

4.2. Case 2: AR Youtopia- System Description

The AR Youtopia system was developed by meeting the core design requirements of the Youtopia tangible tabletop system. This system was created using Unity3D[8] for the core development and ARCore[9] for the tabletop multiplayer AR functionality. All the graphics and code used for this project is open source and can be found in Github[10]. The code can be used for any personal and commercial purpose maintaining the GNU General Public License 3.0[11].The AR system consists of one mandatory part and one optional part that improves technical and user experience. The mandatory part is (1) An AR mobile application; the optional component is (2) A physical printed referential map. The AR application is the most prominent part of the system. The map, on the other hand, is an important part of supporting the first one. The app should also work fine without the absence of the physical map, but the presence of the map emphasizes a referential anchor for both users. The map can be downloaded as a pdf file from the link provided inside the app and print with a regular laser printer and taped together. The core interaction of the system includes starting a multiplayer session, input stamp distributed among users (as roles or non-roles) and then placing and manipulating (i.e. move, rotate, delete) digital stamps to create land-uses. Other interactions involve using the impact tool (shows environment status) and info tool (shows information related to a stamp). I will discuss more on each interaction with corresponding design considerations in the next section. In this section, I am going to discuss the core interaction and system architecture with technical implementations.

The first part of the system is the AR mobile application. The system needs two networked Android/iOS-enabled tablets or mobile devices with at least Android 7.0 (Android Marshmallow) or iOS 9.0 version to run the application. The specs were chosen so that the application is supported in most of the current or three – four-year-old Android or iOS devices promoting affordability. Just like any typical LAN (Local Area Network) multiplayer game, the application starts with one user (e.g. User A) creating a new room and others joining it. The multiplayer functionality is achieved through Google AR Core's

---

[8] https://unity.com/

[9] https://developers.google.com/ar

[10] https://github.com/Shubhra22/TUI-to-AR/tree/master/ARYoutopia

[11] https://choosealicense.com/licenses/gpl-3.0/

cloud anchor[12] technology. After everyone joins the room, "User A" creates a cloud anchor by touching anywhere on the screen. For this application, I used the anchor as an augmented map on top of the physical map. The pose[13] of this anchor is saved to the cloud by the system immediately after it's placed. The pose of an object (game object inside the application) refers to the local position and orientation of the object to the world space. When the map is created, each of the two users will able to see it in their devices, and then they will get a set of input stamps based on their role if applicable. The stamps are basically digital buttons that enable the users to pick a particular land-use type that they want to place in the environment. I will talk more about the stamps and their design considerations in the next section. Now each of the placed land-use types is synced with respect to the cloud anchor to make sure they are sitting at the same position for each user. This part is automatically done by the ARCore SDK. All the object placement dependencies (e.g. sequences of land-uses), and their impact on the environment are derived from the tangible Youtopia tabletop system. My main contribution to this work is the technical development of multiplayer AR applications and design recommendations for AR apps. I will discuss more on them in the next chapter. The diagram shown in Figure 4.11 shows how the system works for two users.

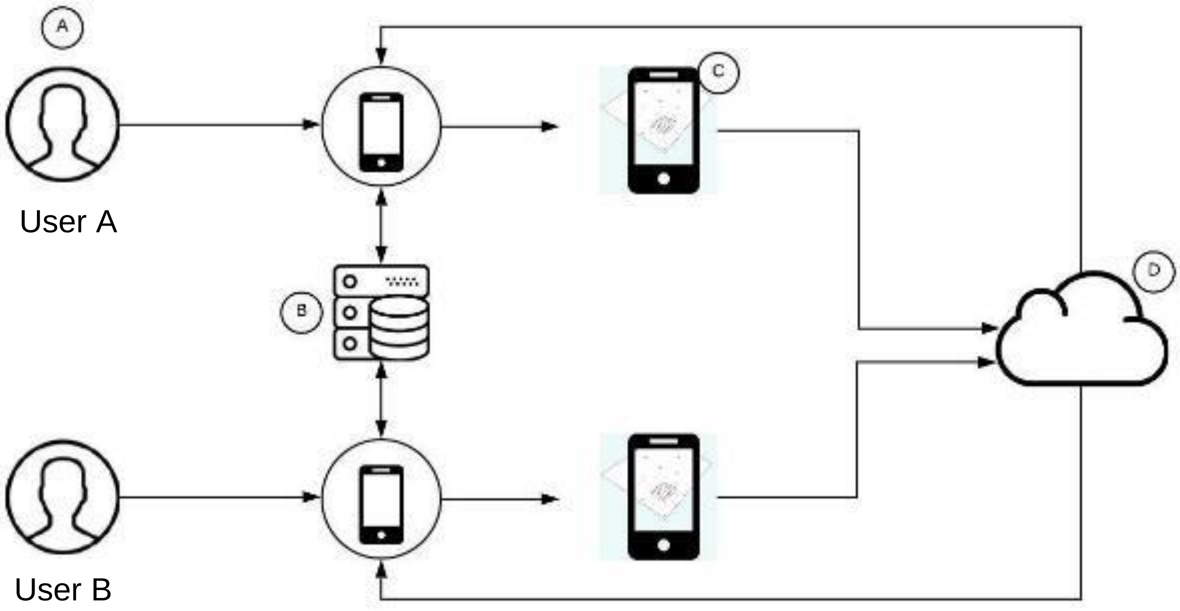


Figure 4.11 System architecture for AR Youtopia – two users.

---

[12] https://bit.ly/2rbjSrq

[13] https://bit.ly/2QytOFZ

A. Physical set up: User A and User B both have their own mobile device that would be later connected in the same network. Both users should be able to physically see, talk and interact with each other. The users should share a common physical map that will later be tracked by the system to place the map anchor. The physical map design is the same as used in Youtopia. For the sake of the prototype, I used only one of the four available maps from Youtopia for this research. The physical map is not technically needed to run the application; however, it provides better ground tracking, resulting in better performance for the application. It also acts as a referential anchor between users promoting better user experience. Users should be standing during interacting with the system so they can move around easily and quickly with respect to the world map in order to explore and place land-use stamps, as most of the objects are generated in world space.

B. Sessions: When the physical set up is done, User A will start a new session (or host the game) from his or her device. Creating a new session refers to two steps (a) creating a new room that other player(s) can join and (b) creating a cloud AR anchor that will work as a referential point to create the multiplayer AR experience. The players matchmaking is done through Unity's multiplayer host service[14]. In step (a), both users get connected on the server, and the application gets synced. This enables both users to see each other's activities (e.g. placing stamp, deleting object, see info or impact tool etc.). In step (b), the host (the user who creates the room) should create a cloud anchor by simply tapping anywhere on the screen as instructed. This cloud anchor is synced to both the users' devices. From this point onwards, all the position and rotation (poses) of each stamped land-use type will be synced through the connected devices. Next, I am going to talk more about the cloud anchor and how I used ARCore to combines this technology to make multiplayer AR happen. Although the ARCore SDK comes with an example of how to use the cloud anchor, I had to write custom code to add cloud anchor compatability with my project's context.

C. To create a cloud anchor, User A should first move the camera and scan the physical map to get enough feature points. The user will be instructed to move

---

[14] https://unity3d.com/unity/features/multiplayer

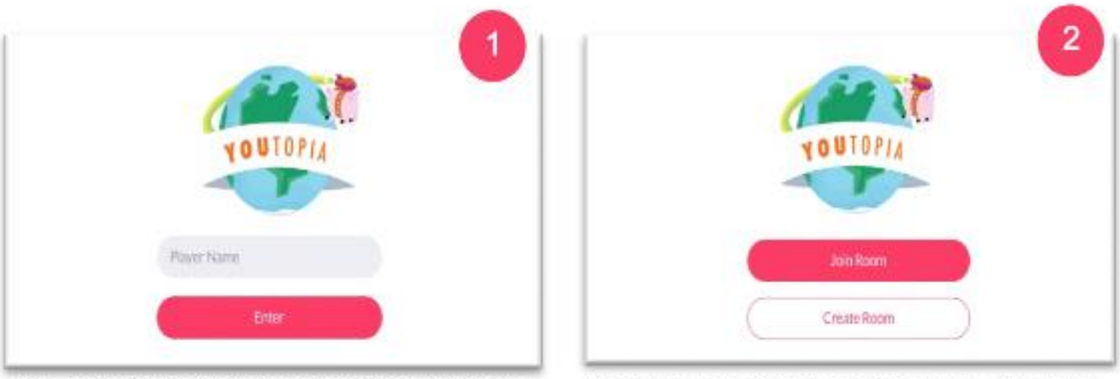
the device and mesh is generated on the portion where the ground plane is tracked. This visual cue(Figure 4.12c) and messages (Figure 4.12a, 4.12b) can help the children to understand that if the system is working or not, Technically, the ARCore plugin takes the feature points and generates a ground plane mesh on top of it. However, the user does not need to know the technology behind this. After the ground plane mesh is created, User A can tap anywhere on the scanned area to spawn the cloud anchor. The cloud anchor is simply an augmented map overlay on top of the physical map. This enables a digital blueprint of the physical map stored in the cloud. In other words, the overlay act as the digital reference point for the AR application that syncs all the objects placed by the users through a cloud service (Google Cloud). After User A places the anchor, its pose is sent to Google Cloud which I will talk about next.

D. The Google Cloud plays a vital role in providing a seamless AR multiplayer experience. All the objects placed by the users are synced through the cloud. For example, after User A hosts the anchor (the digital overlay on the physical map), Google Cloud server takes that information and creates a room number corresponding to that anchor. This is done by collecting enough feature points surrounding the anchor. Other users are synced when they enter this room number in their application. Google Cloud will look for similar feature points and resolve the anchor in all the connected devices. At this point, all the users will be able to see the anchor in their devices. Now, each time a user puts an object on the screen, its pose will be calculated with respect to the anchor and stored in the cloud for other users to see it.

Figure 4.12 (a) Give message if device is moving too fast. (b) Provide feedback if surface is not trackable (c) draw mesh to provide feedback that ground plane is detected.

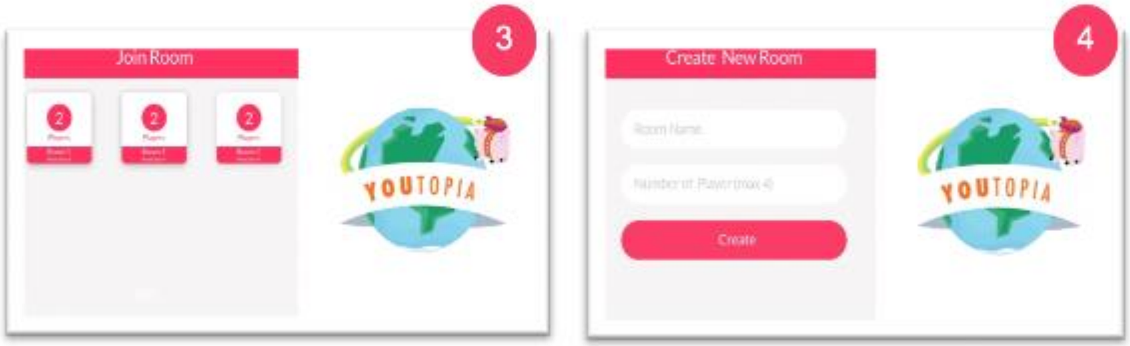
4.2.1. Use Case Scenario for AR Youtopia

In this section, I talk about the process with which the users interact with the system. The screenshots are taken from one user's device. A similar process will be true for other users. Below in Figure 4.13 and Figure 4.14, I have discussed the interaction process with Youtopia AR with appropriate screenshots. The number represents the steps or phases of each interaction.



The application starts with a textbox where the player should put their name and enter the game.

Then the user either joins in any existing rooms or creates new room by simply hitting one of the buttons.



If the user clicks on the Join Room button, the system is redirected to join room panel. Here all the existing rooms are shown, and user can click one of them to join.

If the player had clicked on create room, the system is redirected to create room panel. Here the user can create a room by giving a name and number of players. Maximum four players can join in a single room.

Figure 4.13 User interaction flow for AR Youtopia showing how multiple users connect into the game.

After user creates a room, or join an existing room, she is then redirected to that particular room. Here the user can click on the ready button to start the game. This panel also shows the list of all users currently in the room.

In the game the user can select one of the stamps by clicking corresponding button. The stamp panel follows volumetric UI. The panel can be panned by simply dragging. The panel's position changes with user's physical position.

After selecting a stamp, a user can place it by simply pointing to ground and tap anywhere to spawn that type of object. Tapping on a spawned object shows more option: drag to move, clicking 'x' button to delete, click on the 'i' button to launch information tool corresponding to this object.

Adding more stamps in to the world will eventually start affecting the environment. Users can go to impact tool by clicking the impact button placed in the top right corner of screen and see the impact to the environment due to their decision. Impact tool is popped up on both screens obscuring the game play view and only the user who triggered it can turn it off.

Figure 4.14 Interaction flow for AR Youtopia showing different AR interactions inside the game.

4.2.2. Key Design Features and Rationale

The AR Youtopia system was designed considering the same design requirements gathered from the tangible tabletop Youtopia system. I took all design requirements (DR#1 to 9), as discussed in Chapter 2.2.5, and each of them was applied in the AR system. To promote affordability, the tabletop was replaced with two or more mobile devices with the application installed. In this section, I will discuss how the design features of tangible Youtopia were converted into the AR Youtopia application. I will also discuss what different techniques (if any) I tried to solve a design problem and which final design I choose and why.

R1. Co-dependent access points. The co-dependent access points require more than one input for particular system response. For example, to create a farm, three irrigation stamp inputs are needed near the river, followed by the farm stamp input. The system creates an error message if the user tries to create a farm without irrigation; that is, the system gives error messages if a land-use type is placed in the absence of its prerequisites (Figure 4.15a). This is identical to the tangible system. The AR system strictly follows the same error message, requirements and co-dependent access point design like the tangible system. To precisely place, an object near to a co-dependent object user can use the crosshair interaction (Figure 4.15b) When the crosshair enters into a range of a co-dependent object, the boundary is drawn digitally in the space, so the user knows if he can place the stamp inside that area.

Figure 4.15 (a) Error message provided when a user is trying to create a farm but the prerequisites(irrigation) are missing. (b) Closer look of the circular crosshair used in the project.

R2. Interdependent Access Points. In the tangible Youtopia system, each physical stamp has either a wrench icon (represents developer stamp) or a tree icon (represents resource manager) (Figure 4.16a). Users can select particular roles by picking physical stamps with particular icons (wrench or tree). For the AR version, there were multiple considerations for the replacement of the physical stamp. I chose digital spatial buttons as input stamps (Figure 4.16c). I will discuss the considerations and my chosen design decision later in this section (R8). The assignment of the role is done over the cloud**.** At the start of the game users will be asked to choose a particular role among the two: (a) Human Developer (b) Natural resources manager. The system will then automatically assign corresponding input stamp to the users. The assigned role of a user is presented in a pink colour with all the stamps he/she can access (Figure 4.16b). One user can not access the other user's stamps, which is presented in gray colour with other user's role(Figure 4.16b) In the tangible Youtopia, there were three stamps that are neither belongs to the human developer, nor the resource manager. They are: (a) info tool (2) impact tool (3) delete tool. For the AR version, these tools are replaced with UI buttons and kept separated from the assignment of roles. I will discuss more info, delete and impact tools later on in this chapter (R3 and R7). One thing to note here is that after a user selects one of the roles, the other user will not be allowed to select the same role so that control will be switched off for him.

Figure 4.16 (a)Tangible stamps with wrench or tree representing roles. (b) Digital buttons representing stamps of different roles, pink stands for active role for current user, gray stands for inactive. (c) Spatial UI buttons used to represent stamps in the AR Youtopia.

R3. Pause interaction with reason to reflect. Reflective pauses are important in collaborative platforms, which refers to pausing the interaction to reflect. Similar to the tangible version, the AR Youtopia has two design features that support reflective pauses. The information tool and impact tool, as discussed in Chapter 2.2.5, contain the same content in the AR system as the tangible one. Representation of the information again could be done in either screen space or AR world space. For this interaction, again, I chose the screen space UI to avoid perceptual usability issue discussed in Chapter 2.5. When triggered, the contents of Information or Impact tools are spawned in both devices, and the interactions with the AR world in both devices are paused. To trigger the information tool user can simply tap on a "stamp" button and click on the info icon placed on the top right corner of the stamp button (Figure

4.17a). To trigger the impact tool, a user can click on the Impact button placed on the top right of the screen (Figure 4.17b). One thing to note here is that only the person who triggered the impact tool can turn it off. Interaction is turned off for the other user, so he/she has to depend on the other user.

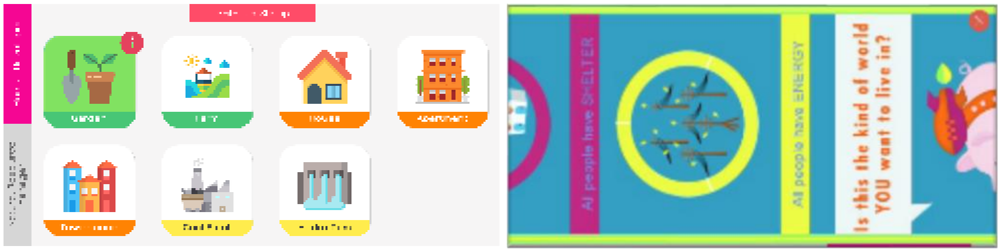


Figure 4.17 (a) User can get access to the info tool by tapping the button on top right corner of a stamp. (b) Contents of impact tool is placed as an on screen scroller UI in both user's screen. The user who invokes the impact tool can only close it.

R4. Activity monitoring of each other. One of the design considerations in tangible Youtopia was that the users could monitor each other's activity. For a physical world, this design could be achieved easily. However, for the AR version of Youtopia, the activity monitoring is quite challenging. One user can see the activity of other users partially through their device camera's field of view. This means, whenever one user moves, places or deletes a land-use stamp that updates in real-time on the other user's device screen. The system also updates the currently selected stamp information to the user's device screen. For example, the tool one user selects is reflected in other user's screens. Also, the impact and info tool's result is reflected in both the user's screen. But,they don't see each other's error messages. They also can not see the other user picking up and hold a stamp but not placing it. This means they can't see a user's intention to act but only the result of an action. That is why the AR Youtopia could not meet the requirement of activity monitoring.

R5. Gaze Monitoring of each other's gaze. Unlike tangible Youtopia, the AR version cannot promote gaze monitoring between users. One of the main reasons for this is that all the interactions take place on the device screen. As a result, the users will mostly be focused on the device screen. This problem could be solved through head-worn AR experience (e.g. head-mounted displays like Hololens[15], Magic Leaps, [16] etc.) , but that would not meet the accessibility requirements. So, similar to activity monitoring the AR Youtopia could not meet this requirement for gaze monitoring. That said, it is possible for one user to look at what the other is attending to on that user's screen and vice versa, but I did not solve this design requirement explicitly.

R6. Referential anchors. In the tangible version, the tabletop itself with a screen space map acted as the referential anchor. In the AR version, the map is replaced with a physical printed map. While technically, the system can work fine without the map, it provides a common shared input space for the user. However, to make the map interactive, an augmented terrain is generated on top of the physical map (Figure 4.18). The augmented terrain is just an augmented image of the physical map. Different elements on the map get changed based on different user's input. All the stamps can only be used on top of this augmented terrain, i.e. the interactive map. So, in the AR Youtopia, the digitally augmented map and the printed physical map act as referential anchors.

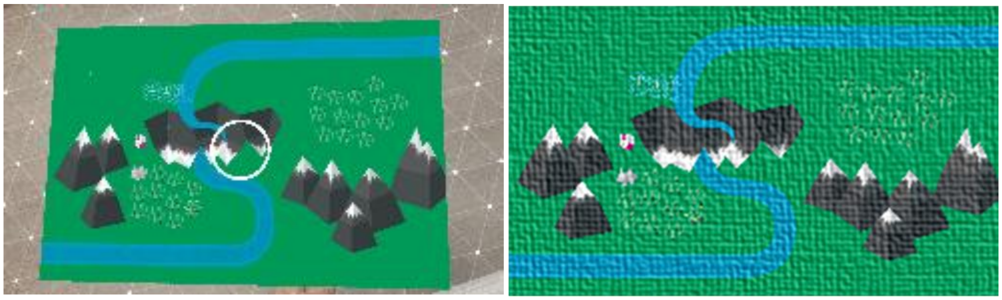


Figure 4.18 (a)Digitally augmented map will act as the referential anchor for both player. The circle in the center is the crosshair for interaction. The white meshes in the background provide users information about tracked ground. (b) The image of one of the maps that the user need to print.

R7. Objects of negotiation. In the tangible Youtopia, there were three tools (physical stamps) that the users had to share promoting positive negotiation. They are: (1) info tool: helps to get information of a particular stamp; (2) impact tool: shows overall status of the environment such as pollution level, food level, shelter level and energy level ; (3) erase tool: helps to erase a land-use items places in the map. In the AR Youtopia version, these tools are replaced with

---

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

[16] https://www.magicleap.com/

UI buttons. To promote negotiations between users, these can only be used by one user at a time. The user currently using a tool can share it with the user

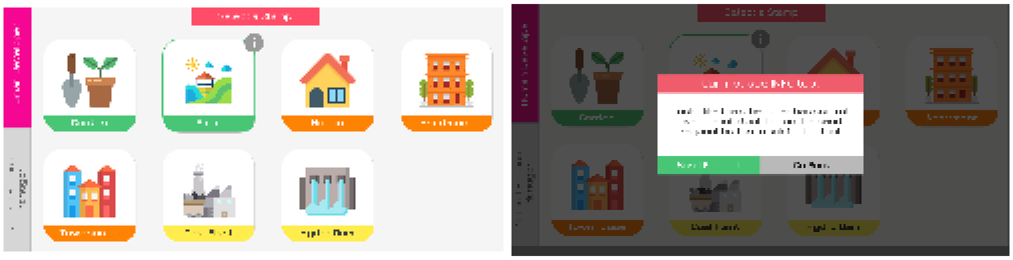


Figure 4.19 (a) Info tool turned into gray colour,representing inactive. (b) User gets an error message if the inactive info tool is tapped.

upon request. The tools change their colour from pink to gray when the user does not have control over it. (i.e. other user has it.) When tapped on an inactive button, it shows an error message as well as a request button, which can be used to request the tool to another user. Figure 4.19(a) and 4.19(b) shows an example of this flow with the info tool.

R8. The number of input objects. Similar to the tangible tabletop, the AR system has 13 different land-use input objects, as well as 3 general-purpose tools (Info, Impact, Eraser). However, for reasons explained below, the physical stamps were replaced with digital spatial buttons. First, I considered using small tangible stamps, which can be put on in the fingers, like finger puppets. While this could introduce a fun way of interaction with the system, this might cause multiple problems such as (a) holding the tablet while wearing the stamps can result in bad tracking (Usability Issue 5 - Chapter 2.3 page 23-24), (b) changing a stamp from a finger, since there are 13 stamps, and we only have 10 fingers and (c) remembering which stamps are associated with which finger (Usability Issue 3,7 - Chapter 2.3 page 23-24). For example, changing a tangible stamp

Figure 4.20 Selected stamp with gaze interaction and with radial progress bar to show the selection status.

from a finger is a two-hand interaction, so the user would need to put down the tablet, which may lead to bad/lost tracking of the system. In addition, this adds the necessity of creating or purchasing finger stamps or finger puppets. So, rather than using tangible stamps, I decided to use digital buttons as a representation of the stamps. At first, I tried with on-screen buttons. However, it takes spaces on the screen and blocks the AR terrain area. Besides, it needs the children to tap on the buttons which might be a bit challenging for children as they need to hold the tablet with one hand and tap on the screen with the other hand. So, I decided to use a spatial UI design with a gaze system to select a stamp. The user can move their camera vertically to see the interactive stamp panel and use the gaze to select a particular object. When the gaze is placed on top of a button UI, the colour is changed to green, and a radial slider shows the current state of selection, representing how much time left before the autocompletion of the interaction(Figure 4.20). The user can either tap anywhere on the screen to complete the interaction or wait for the gaze to complete the selection. This is one of the most innovative aspects of the AR Youtopia solution.

R9. **Use of stamp one child at a time.** When a user selects a stamp, the control to the other user is turned off so that two users cannot use stamps at the same time. This process is automatically handled by the system and forces the user to wait for the other user's input. All stamps of the inactive users turn into gray representing the stamps are not usable (Figure 4.21a).The system also provides a message to the user describing that another user is using the stamp and he/she has to wait until the other user is finished (Figure 4.21b).

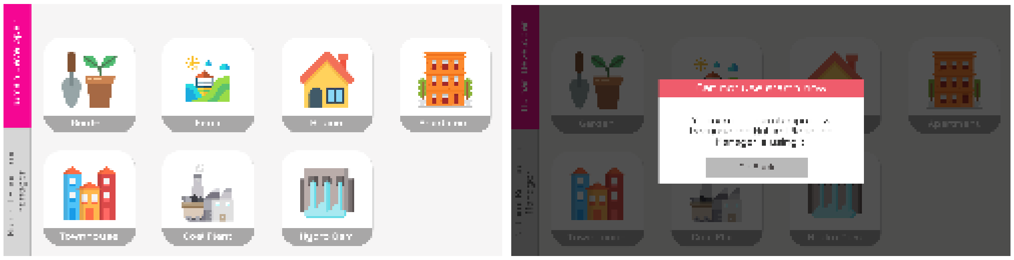


Figure 4.21 (a) All the stamps turns into gray if other user is using the stamp. (b) If user tap on an inactive stamp it shows an error to get a clear message to the user.

Chapter 5.Technical and Design Implications

In this chapter, I will summarize the problems I solved while developing the two different AR applications discussed in this thesis. I will discuss the main technical and design contribution to these projects and discuss how my findings may generalize the development of similar systems in this area.

5.1. Case 1: AR PhonoBlocks Technical and Design Implications

1. **Use of Deep Learning** Some previous works have used AR technology to enable language learning for early ages. (Fan et al., 2018b; Gandolfi et al., 2018; Mahadzir and Phung, 2013). One common technical problem exists in those works, which is the challenge of making the tracking stable. This problem can be solved by using complex markers (Fan et al., 2018b). However, this solution only works seamlessly for one to four-letter words. Moreover, letters like 't' or 'l' are really difficult to detect reliably with the system as they don't have enough surface area to attach a complex marker. From a design perspective, market-based approaches have some other issues as well. The process of making and training different sets of markers is time-consuming. Besides, the end-user does not have control over the marker design relative to their needs. As a result, they cannot customize it to get better performance. Also, printing the markers and sticking them in the physical letters requires some degree of skill (e.g. 3D modelling for laser cutting software), equipment (laser cutter), workmanship (refining cut letters and attaching appropriate stickers) and technical know-how (setting up the system for marker set) to make it work. The marker-less approach solves these problems by taking custom markers out of the picture. One of my main contributions to AR PhonoBlocks is the development of a deep neural network to detect alphabetic letterforms without any need for markers. With a deep convolutional neural network (CNN), I was able to detect all the letters with an accuracy of 96%. The train and test sets were taken as 70% and 30% of the whole data set, respectively. The accuracy was calculated based on the testing data set. From the test set, I took the number of correctly predicted letters and divided them

by the total number of predictions. More information on how the model was trained and tested can be found in the Github link[17] of this project. For some of the letterforms, the detection performs inaccurately if the letters are close to identical. For example, letters that have a superscript dot (i.e. diacritic) such as the dots on a lowercase ‘i’ or ‘j,’ for the system to accurately identify, the dot must be distinct from the line aspect as shown in Figure 5.1b. If they are merged (Figure 5.1a), the system fails to identify the letter. There are already some popular OCR (Optical Character Recognition) plugins like the Tesseract[18], Google Cloud[19], Microsoft Cognitive API[20] , and so on, but they are either pay for use (google cloud and MS cognitive API) and/or not customizable to fit in my research area. For example, when trying the Tesseract wrapper for unity works really good for the typed letters (such as a paragraph of text in a book) but fails to detect 3d letter shapes like the one I used in my project. The detection is also poor for single letters than a sentence. My CNN model works both for printed (2d) letters and 3d letter, can be customized by using transfer learning (discussed later) and should work on any sans-serif typeface fonts. Using this same model, other researchers in this field would be able to develop similar applications related to 3D optical character recognition for letters or other similar forms. Using transfer learning, this model can also be used to develop similar applications for other languages. Transfer learning is the process of using the same CNN model and training it with a

Figure 5.1 (a) Similar looking letter detection can be hard for the CNN model like ‘i’ and ‘l’ (b) Distinct dot of the letter ‘i’ from line aspect makes the letter detection accurate.

---

[17] https://github.com/Shubhra22/TUI-to-AR

[18] https://github.com/tesseract-ocr/tesseract

[19] https://cloud.google.com/vision/docs/ocr

[20] https://bit.ly/2NngIcD

different dataset to get a different result. For example, to make my CNN model workable with another language the training dataset should contain images of each letter or number of that particular language. Then the developer needs to feed this training set into the CNN model and change the last layer of the model which provides the final output.

2. **3D Augmented Colour Cues** As discussed in Chapter 2, real-time colour coding or dynamic colour cues can enhance the language learning process of students by directing attention to key processes relevant to learning (Antle et al., 2015). Although many previous works focus on letter tracing and 3D AR representation of whole words, there has been less work regarding real-time 3D colour augmentation that improves the process of learning by directing attention (or improving other learning processes) or operates at the level of a letter versus the whole word. Fan et al. (Fan et al., 2018b) have demonstrated a marker-based AR version of PhonoBlocks with 2D colour mapping on the screen. However, in their demonstration, the dynamic colour changes for each letter are not displayed over top of the physical letterforms, just on the screen in a prespecified location. In my work, I have managed to come up with a proof-of-concept solution for projecting colours on the 3D letterforms in real-time, which meets the requirement to use colour to draw the user's attention to the letter changes and letter-sounds correspondences. The standard approach for colour or object augmentation on a markerless object is made by tracking a lot of feature points on an object and augmenting data on top of it. However, for low feature objects like a single alphabetic letter form, object tracking does not work because of the absence of enough tracking information [21]. So, tracking the letterforms with trained data (with the help of CNN) and then augmenting through contour analysis and then changing each relevant pixel contour value. My proof-of-concept (code from Github) and approach can be generalized and used by those to developing similar augmented reality applications where real-time colour augmentation and segmentation is necessary. The model is best suited for alpha-numerical character shapes (e.g. numbers, arithmetic operations, punctuation, other alphabets etc.) in addition to other letter sets

---

[21] https://library.vuforia.com/articles/Training/Object-Recognition

(e.g. capitals). The model, however, will not be able to train complex 3D shapes from the real world (e.g. table, chair, cube, etc.).

All the design decisions taken for AR PhonoBlocks are derived from the marker-based AR PhonoBlocks designed by Fan et al. (2018). In this version, the marker-based letters were replaced with markerless letters to make it less custom hardware dependent. Other than that, there is no design contribution made for this project.

5.2. Case 2: AR Youtopia Technical and Design Implications

1. **Multiplayer Augmented Reality** Some previous researchers have shown potential uses of AR as a collaborative learning platform (Kaufmann, 2003 ; Alhumaidan et al., 2018). I have also discussed works that have used multiplayer AR to promote collaborative learning (Alhumaidan et al., 2018; Shin et al., 2018). However, there is a big gap between these works and currently available AR technology. Most of these works include expensive specialized hardware or obsolete tools and technologies. As discussed in Chapter 2, the existing HCI works in the field of collaborative learning have not adopted affordable AR technologies. As a result, there has been knowledge and technical gap for researchers in this field related to how to use this technology in an affordable approach for collaborative learning applications. One of the core contributions of my work for collaborative learning is the implementation of multiplayer AR experience with available, affordable technologies. While multiplayer AR experience is quite commonly available for high-end AR technologies such as AR headsets, they are not affordable. On the other hand, hand-held AR is much affordable but less stable. There are different commercial solutions for hand-held AR multiplayer apps. For my work, I have used the ARCore cloud anchor to provide multiplayer experience. Both ARCore[22] (by Google) and ARKit[23] (by Apple) SDK can be used with AR cloud technology to make multiplayer AR experience. I have integrated the Google's

---

[22] https://developers.google.com/ar

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

Cloud Anchor with ARCore SDK to achieve the multiplayer AR experience for AR Youtopia. My implementation of the multiplayer AR technology to promote collaborative learning can provide proof-of-concept knowledge (the documented code of the application) which has been put on GitHub[24] to aid others develop a similar collaborative AR application.

2. **AR Design Guidelines** While developing the AR version of Youtopia, I have considered usability issues for handheld AR applications derived from different researchers over time (as discussed in Chapter 2.5). Considering these usability issues, I have come up with some design guidelines for AR applications, especially for children. While these decisions have not been tested with real users, they are adapted from different findings from prior research. One thing to note here is the design guidelines are applicable for table scale AR applications and may or may not apply to room-scale or world-scale AR applications.

    a. **Crosshair vs Finger Interaction** A common interaction in any handheld AR application is spawning or selecting an augmented object in the real-world environment. Currently, there are two widely used interaction techniques to achieve this. (1) *Finger based touch interaction* where the user needs to touch on the screen to spawn something in that location in world-space. (2) *Crosshair interaction* where there is a crosshair in the middle of the screen which moves according to the device's movement. The finger interaction should be used in places where the screen size is small (so any position in the screen is reachable while holding it with two hands), and the user does not need to place the objects in any precise location. For a tablet game, this kind of interaction is not recommended as it might be hard for the users, especially children, to hold the tablet in one hand and tap on a precise location on-screen. Holding a tablet on one hand can reduce the precision of the interaction point or loose tracking information. So, in this case, designers might consider using crosshair-based interaction. For more precise positioning (for example, in AR Youtopia, the user needs to place a garden close to an unused irrigation system), it's better to use crosshair-based interaction as it gives the user the ability to hold the tablet with both

---

[24] https://github.com/Shubhra22/TUI-to-AR

hands and precisely select the target where to place the land-types. Researchers have argued that crosshair based interaction is slower in terms of selecting or moving objects (Radu et al., 2016). However, many modern handheld AR apps are adapting the crosshair-based interaction (i.e. Apple's measure[25] app) where they make the crosshair larger and snap it to the closest object of interaction. This solves the concern of time and precision mentioned by Radu et al.

Design Recommendation: Use crosshair/reticle-based interaction while developing apps for children, especially if the interaction needs to be precise.

b. **Spatial vs Screen space UI** While designing for AR, it's important to make sure that the screen contents do not obscure users' view of the world. This is where spatial UI or volumetric UI comes into play. Spatial/volumetric UI are UI components that are placed in the 3D world space, just like the augmented objects. As users tend to be focused on the AR scene, this kind of interaction can be intuitive during the gameplay. Spatial UI components can be helpful in holding small interactions or information. For example, providing short interactions (e.g. edit, delete, info buttons, etc.) through spatial UI for individual augmented objects can facilitate object manipulation. Another use of spatial UI is presenting information that is frequent. For example, showing an error message in spatial UI will not obscure users' views and at the same time, fulfills its purpose. Screen space UI, on the other hand, are UI components that are placed on the device screen. This kind of UI can block users' viewport to the world but has some specific advantages for use cases as well in AR design. For example, presenting text-heavy information in spatial UI might make the information hard to read. So, it's better to use on-screen UI to present information-heavy content. There are other places where on-screen UI can be beneficial. For example, controls that a user will need frequently (e.g. selecting a stamp in Youtopia) should be placed on the screen. One thing to consider here is to use drawer UI so the user can collapse or expand

---

[25] https://en.wikipedia.org/wiki/Measure_(Apple)

particular controls to avoid viewport blocking. Another situation where on-screen UI can come handy is intentional pauses in the interaction. For example, in Youtopia, I have used this technique to present the impact tool (i.e. user can check the state of the world through impact tool).

Design recommendation: Use on-screen UI for quick action interactions and intentional pauses. Use spatial UI for micro-interactions and interactions related to AR object manipulation.

c. **Visibility of Object Status** Visibility of an object's status is important for the user to perform any changes to the object. For example, if the user does not know which object is currently selected, it may be hard to manipulate that object as she may forget its spatial position. Besides, if the user wants to move an object beyond the camera's current viewport, he might lose the object, among others. Providing some sort of spatial marker is important for the user to understand that the object is currently selected. If there is any sort of radical/cursor interaction, it's also important to show which objects are getting triggered by the cursor. Another scenario can be something like when a user tries to move an object too close to another object; there should be some kind of indication that restricts them from doing so. Finally, it is a good idea to show the status of the detected plane and the orientation of the cursor with respect to that. For example, the cursor should be dynamically changing its orientation while changing from a horizontal plane to a vertical plane or vice versa.

Design recommendation: Show feedback on the object status during the interaction. Use spatial markers to show the object status.

d. **The Dynamic Scale of Spatial Interfaces** One problem of spatial UI is that the size of a UI changes with its distance from the device. This can lead to the problem of unreadable text (Chapter 2.5 Issue 8), inducement of extra cognitive load (Chapter 2.5 Issue 2) and continuous movement in-between space (which can be immersive, but tedious for children). Besides, as the UI size decreases, the interaction precision problem (Chapter 2.5 Issue 1.6) arises for child interaction. This issue can be solved by dynamically scaling the spatial UI components with respect to its distance from the device. One thing to consider here though, is maintaining a threshold or maximum size of the interface elements. Otherwise, this might be aesthetically unpleasing and obscure the viewport for the user.

Design recommendation: Consider scaling spatial UI dynamically based on their distance from the device. While picking a scale, make sure (1) its big enough to easily interact; (2) it's not blocking the user's viewport.

Chapter 6. Discussion

In this section, I discuss the implemented proof-of-concept prototype of different features, adapted from the two tangible applications (PhonoBlocks and Youtopia). I have ensured the affordability of both systems by eliminating custom hardware, compatibility of the app with all AR supported phones and made design decisions based on prior knowledge adapted to my solution space that may make the app easy to use. I have made the systems scalable by using the up to date tools, plugins and SDKs which are compatible with most of the current devices and platforms. Below, I am going to define my main contributions, how they can be generalized to similar other applications and the limitations of my work.

6.1. Contribution

In the literature review discussed in Chapter 2, I have talked about previous research works done in the field of AR language learning and collaborative learning. From the existing reading applications, I have found that researches have been mostly dependent on custom hardware (Juan et al., 2010), lacks design guidance (Silva et al., 2013), does not help students to understand the letter-sound correspondence as they focus on whole words (Boonbrahm et al., 2015) and none of the application used markerless AR which makes the application dependent on custom markers. For the AR PhonoBlocks, my main contribution is the machine learning model, which used a convolutional neural network (CNN) to detect 3D letterforms for the AR PhonoBlocks project. The application removes unwanted noise, including shadows from the tracking and recognizes letters in 0.04 seconds. The CNN model can be generalized by developers and researchers to develop similar kinds of applications where the developer needs to track different alphabetic physical symbols. As it worked with English letters, the model can be adapted for other languages by training with that language's alphabet through transfer learning[26]. Obviously, the model needs to be fed by training images of that language. The training dataset should contain images of each letter or number of that particular language. Then the developer needs to feed this training set into the CNN model and change the last layer of the model which provides the final output. There are many

---

[26] https://en.wikipedia.org/wiki/Transfer_learning

OCR plugins, SDKs and deep learning models available in marker and research community, but there is a lack of models that detect 3D letter shapes. Besides, most of the traditional OCR applications fail to work only for a particular language, not easily extendable to other language and sometimes does not work on different font or geometry shape of the object. The use of deep neural networks can fill these gaps. Knowledge from this work can be helpful in implementing different literacy acquisition applications such as early reading, remedial adult literacy, ESL, etc. I will provide the CNN model and datasets as opensource so that anyone can use this for their work. There is also scope for developers to adapt the code to develop real-time OCR application. The code is opensource and modular, so a developer can either use the CNN model I have used (for English OCR) or just replace their own trained model to make it work.

For the AR PhonoBlocks project, I also implemented a colour projection technique on markerless letters. Although this feature is not completely stable and the performance can vary from one device to another (detailed discussion in the limitation), the technique as it currently stands can provide knowledge to develop an application where real-time colour augmentation is not necessary. Apart from the technical contributions, my work results can be used specifically to create AR PhonoBlocks research prototypes that serve as a scalable solution for further research in this area. Although the system is not stable yet in terms of colour augmentation, with more research, it has the potential to imply this knowledge to real-world applications and larger-scale adoption in the future.

The second prototype I developed is a collaborative AR application based on the tangible tabletop system, Youtopia, which was designed as a collaborative learning experience for two users. From the existing works on collaborative AR learning, I have noticed that most of the applications have barely used the state of the art AR techniques which has made the solutions obsolete (Regenbrecht et al., 2002; Shin et al., 2018) and becomes hardware dependent (Alhumaidan et al., 2018) in most cases. I used some established design guidelines for collaborative learning (Antle and Wise, 2013) for the prototype, but most of the discussed works are missing these. For example, these works are not designed based on guidance for collaborative learning. Also, the works are missing a scalable networking solution for collaborative AR experience. My main contribution to this project is the implementation of multiplayer AR experience using the power of AR cloud anchor, a new feature included in the ARCore plugin by Google. This contribution will be primarily useful for UX designers and researchers who work on collaborative learning or multiplayer games. Because existing works in this research area have hardly used the power of modern mobile AR technology its often too expensive for researchers to work on this field. My work can provide knowledge to develop affordable AR applications in the research and development area of collaborative learning. It can also be used to inform how tangible multiplayer applications can be transformed into AR applications utilizing cloud anchor technology. The technical solution can be applied to similar collaborative applications such as architectural and industrial 3D data visualization, medical data presentations, etc. I have also come up with some basic design guidelines for handheld AR applications specifically for children which I discussed in Chapter 5. The design guidelines can provide knowledge for new researchers and designers on this field to design common UX for tabletop AR (as discussed in Chapter 4) such as onboarding, visibility of system status, crosshair-based interaction, etc. My major goal for these prototypes is to transfer design ideas from one technology to another. That is why I have only focused on primary validation by figuring out whether the system works or not, and if it works well enough. This has provided knowledge on whether a design is transferable from a tangible system to an AR system. However, the next steps should be validating the user experience of the application through a secondary validation technique. In the next section, I am going to talk about next steps if someone was to carry on this work and the limitations of my current systems in detail.

6.2. Limitations and Future Work

In this section, I discuss the main limitations of my research, including technical and design limitations. I am also going to discuss how I plan to overcome these limitations in my future work.

1. *Technical Limitations and Future Plan* – There are still some technical limitations of both the applications which need advance computer vision techniques to resolve. For PhonoBlocks AR, the main technical limitation is the time it takes to overlay the augmented colour to the letter blocks. Although the tracking is accurate and smooth, the colour augmentation part is blurry, too much light-sensitive, inconsistent and the augmented image resolution is low. Currently, the colour augmentation is done by contour colouring (discussed in Chapter 4) at runtime which runs at 22 frames per second in a Samsung Galaxy Tab S4 (2018) model. It causes a small delay during the colour augmentation process in the device. This

problem can be solved by using Multithreading or running the image processing code in the GPU using Unity's shader programming. However, Unity's current Multithreading code pattern does not directly support OpenCV. So, this is currently challenging but might be easier with future versions of Unity. Another problem of PhonoBlocks AR is that the system needs a tablet stand for better tracking performance (to avoiding shaking). Although this is not a big issue, finding a good tablet stand that does not block the user's interaction space can be hard. Most of the tablet stands in the market blocks the user's interaction space, and the one with enough interaction space sometimes is too shaky. For this work, I used this tablet stand[27] from Amazon, which gives a lot of space for interaction. Finally, different lighting conditions (e.g. shadows, exposer, etc.) can hamper the performance of the CNN model resulting in poor accuracy as too much light or shadow will add more noise to the input image. Although the system currently handles unwanted shadow and noise by using image filters (gamma correction, gaussian blur), more work on shadow/noise detection and removal is necessary for better user experience in different lighting conditions. At this time these are some of the limits to using mobile AR technology.

The Youtopia AR application is stable in terms of tracking; however, it can also be affected by the lighting conditions. The main technical limitation of Youtopia AR is that the system cannot remember the spatial information of the surrounding environment, which can cause loose tracking, no occlusion, and less immersion. This is because the AR Cloud does not support spatial mapping, so if the player who hosted the AR anchor accidentally closes the app, both the user's game progression will get lost. This can be solved by including spatial mapping API to the application like 6d.ai[28]; however, these APIs are still in beta (not stable) and not available for all platforms (iOS and Android). Hopefully, with the improvement of technology, these will be more stable and accessible to all platforms. Another problem is that the application does not support occlusion which means it will not occlude the AR content from the real-world. This problem can be solved by the

---

[27] https://amzn.to/2r78zAe

[28] https://www.6d.ai/

recent AR Kit2 API's human occlusion feature; however, it is not accessible in Android phones.

2. *Design Limitations and Future Plans* – Both of the applications discussed in my research were evaluated through the primary validation technique of technical HCI, which is making different technical designs workable. However, I have not tested any of them with actual users yet. This is a big limitation for both of the applications because there may be usability issues with the technical implementations. For example, there is a design limitation for AR PhonoBlocks which is the system cannot provide students with any error information if the right letter is placed in a wrong orientation. Also, two of the design requirements of AR Youtopia were unable to fulfill. Future work should include a usability test for both of the applications, which can answer whether or not this design limitation affects the overall performance of the user experience. Before testing the applications with children, I plan to conduct usability testing with 10 to 15 SFU students aged 19 to 30. The reason behind this is to get an overall idea of how easy or hard the system is for adults to interact with and then tune particular pieces accordingly for young children. Also, conducting a study with children is time-consuming, so getting some initial feedback from adult participants can expedite the process. The usability measure should include error count, error severity, and a satisfaction rating of the system.

   Some of the limitations (spatial awareness, gaze monitoring etc.) of my thesis could possibly be overcome by using advanced AR headsets (e.g. HoloLens[29], Magic Leap[30]). AR headsets have recently gained in popularity and we might expect to see more compact versions of the headsets (ex. Apple glass[31]) in next four or five years. When the technology is more broadly available, it may be worth exploring the usability of mobile AR vs AR headsets for the applications in this thesis. However, at this time, there were multiple reasons why mobile AR was considered for this thesis. The main reason was that mobile devices were affordable and broadly available and AR headsets were not. In addition, mobile devices are much more portable than the current AR headsets on the market. AR headsets are also large in size and may cause motion sickness, which might not

---

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

[30] https://www.magicleap.com/

[31] https://www.tomsguide.com/news/apple-glasses

be a good experience for the users (children). However, without doing a usability test these claims cannot be validated. Therefore, it might be a good idea run a comparison between these technologies to find out more in the future. Again, while mobile AR technology has some limitations at this time, it was a better choice for this thesis work because it is affordable and broadly available compared to other AR technologies.

6.3 Conclusion

With the enhancement of computer vision and mobile phone hardware, AR technology is getting more affordable. This has provided an opportunity to provide scalable and affordable solutions for many existing applications. AR applications in learning are also gaining popularity rapidly in the research community. I have discussed two different learning applications that were transformed into table scale AR using tablets while maintaining their core design requirements and considerations. Most of the design requirements from tangible applications were transferred in a new context. A few remained unsolved due to their dependency on tangibility and the current state of AR tools and technology.

The applications that I presented in my thesis provide technical knowledge through the proof-of-concept prototypes I created that may be useful for the design and development of multiplayer AR applications. In particular, those created by adding basic computer vision techniques into the AR applications for better tracking (specifically for OCR based application) and design considerations for AR applications for children. I discussed what different kinds of challenges I faced throughout the process and how some of them could be solved, and some could not. I have also discussed the current limitations of the system, both from a technical and design perspective. The main technical challenges for AR PhonoBlocks were to (a) detect the letters fast enough to make them a smooth user experience, (b) overlaying the high-resolution colour augmentation, and (c) avoiding camera shake. Challenge “a” was solved by using the CNN model and “c” was solved by using a stand. However, the challenge “b” is solved partially by image contouring techniques and need to be refined. All the design requirements derived from tangible PhonoBlocks were fulfilled in the AR PhonoBlock system. The main technical challenge for AR Youtopia was to make the collaborative AR work through multiplayer networking, which was achieved by the Google Cloud Anchor and Unity’s multiplayer services. Most of the design requirement was fulfilled in AR Youtopia, but DR4 (activity monitoring) and DR5 (gaze monitoring) could not be achieved through the handheld AR solution. Finally, I have discussed the areas that can be improved to move forward in the research community. In general, I hope my research promotes the accessibility of AR technology and opens a door for further inquiry on affordable AR tools for education.

References

Ahlström, B., Lenman, S., Marmolin, T., 1992. Overcoming Touchscreen User Fatigue by Workplace Design, in: Posters and Short Talks of the 1992 SIGCHI Conference on Human Factors in Computing Systems, CHI ’92. ACM, New York, NY, USA, pp. 101–102. https://doi.org/10.1145/1125021.1125103

Alhumaidan, H., Lo, K.P.Y., Selby, A., 2018. Co-designing with children a collaborative augmented reality book based on a primary school textbook. International Journal of Child-Computer Interaction 15, 24–36. https://doi.org/10.1016/j.ijcci.2017.11.005

Antle, A.N., 2013a. Exploring how children use their hands to think: an embodied interactional analysis. Behaviour & Information Technology 32, 938–954. https://doi.org/10.1080/0144929X.2011.630415

Antle, A.N., 2013b. Research opportunities: Embodied child–computer interaction. International Journal of Child-Computer Interaction 1, 30–36.

Antle, A.N., Fan, M., Cramer, E.S., 2015. PhonoBlocks: A Tangible System for Supporting Dyslexic Children Learning to Read. ACM Press, pp. 533–538. https://doi.org/10.1145/2677199.2687897

Antle, A.N., Wise, A.F., 2013. Getting down to details: Using theories of cognition and learning to inform tangible user interface design. Interacting with Computers 25, 1–20.

Antle, A.N., Wise, A.F., Hall, A., Nowroozi, S., Tan, P., Warren, J., Eckersley, R., Fan, M., 2013. Youtopia: A Collaborative, Tangible, Multi-touch, Sustainability Learning Activity, in: Proceedings of the 12th International Conference on Interaction Design and Children, IDC ’13. ACM, New York, NY, USA, pp. 565–568. https://doi.org/10.1145/2485760.2485866

Antle, A.N., Wise, A.F., Nielsen, K., 2011. Towards Utopia: Designing Tangibles for Learning, in: Proceedings of the 10th International Conference on Interaction Design and Children, IDC ’11. ACM, New York, NY, USA, pp. 11–20. https://doi.org/10.1145/1999030.1999032

Azuma, R., 2004. Overview of Augmented Reality, in: ACM SIGGRAPH 2004 Course Notes, SIGGRAPH ’04. ACM, New York, NY, USA. https://doi.org/10.1145/1103900.1103926

Azuma, R.T., 1997. A survey of augmented reality. Presence: Teleoperators & Virtual Environments 6, 355–385.

Berryman, D.R., 2012. Augmented Reality: A Review. Medical Reference Services Quarterly 31, 212–218. https://doi.org/10.1080/02763869.2012.670604

Billinghurst, M., 2012. Augmented reality in education, new horizons for learning. 2002.

Billinghurst, M., Kato, H., Poupyrev, I., 2001. The MagicBook: a transitional AR interface. Computers & Graphics 25, 745–753.

Boonbrahm, S., Kaewrat, C., Boonbrahm, P., 2015. Using Augmented Reality Technology in Assisting English Learning for Primary School Students, in: International Conference on Learning and Collaboration Technologies. Springer, pp. 24–32.

Carmigniani, J., Furht, B., Anisetti, M., Ceravolo, P., Damiani, E., Ivkovic, M., 2011. Augmented reality technologies, systems and applications. Multimedia tools and applications 51, 341–377.

Chen, C.-M., Tsai, Y.-N., 2012. Interactive augmented reality system for enhancing library instruction in elementary schools. Computers & Education 59, 638–652.

Cheng, K.-H., Tsai, C.-C., 2013. Affordances of augmented reality in science learning: Suggestions for future research. Journal of science education and technology 22, 449–462.

Druin, A., 1999. Cooperative Inquiry: Developing New Technologies for Children with Children, in: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI '99. ACM, New York, NY, USA, pp. 592–599. https://doi.org/10.1145/302979.303166

Engelbart, D.C., William, K., 1968. English (1968).". A Research Center for Augmenting Human Intellect." In: Proceedings of the December 9–11.

Fan, M., Antle, A.N., Cramer, E.S., 2016. Design Rationale: Opportunities and Recommendations for Tangible Reading Systems for Children, in: Proceedings of the The 15th International Conference on Interaction Design and Children. ACM, pp. 101–112.

Fan, M., Antle, A.N., Hoskyn, M., Neustaedter, C., 2018a. A design case study of a tangible system supporting young English language learners. International Journal of Child-Computer Interaction 18, 67–78.

Fan, M., Antle, A.N., Hoskyn, M., Neustaedter, C., Cramer, E.S., 2017. Why Tangibility Matters: A Design Case Study of At-Risk Children Learning to Read and Spell, in: Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems. ACM, pp. 1805–1816.

Fan, M., Antle, A.N., Sarker, S., 2018b. From Tangible to Augmented: Designing a PhonoBlocks Reading System Using Everyday Technologies, in: Extended Abstracts of the 2018 CHI Conference on Human Factors in Computing Systems. ACM, p. LBW555.

Gandolfi, E., Ferdig, R.E., Immel, Z., 2018. Educational opportunities for augmented reality. Second Handbook of Information Technology in Primary and Secondary Education 1–13.

Hornecker, E., Dünser, A., 2008. Of pages and paddles: Children's expectations and mistaken interactions with physical–digital tools. Interacting with Computers 21, 95–107. https://doi.org/10.1016/j.intcom.2008.10.007

Hudson, S.E., Mankoff, J., 2014. Concepts, values, and methods for technical human–computer interaction research, in: Ways of Knowing in HCI. Springer, pp. 69–93.

Ishii, H., Kobayashi, M., Arita, K., 1994. Interactive design of seamless collaboration media. Communications of the ACM 37, 83–98.

Ivanova, M., Ivanov, G., 2011. Enhancement of learning and teaching in computer graphics through marker augmented reality technology.

Jayant, C., Ji, H., White, S., Bigham, J.P., 2011. Supporting blind photography, in: The Proceedings of the 13th International ACM SIGACCESS Conference on Computers and Accessibility. ACM, pp. 203–210.

Johnson, L., Adams Becker, S., Cummins, M., Estrada, V., Freeman, A., Hall, C., 2016. NMC Horizon Report: 2016 Higher Education Edition. The New Media Consortium, Austin, Texas.

Juan, C.M., Llop, E., Abad, F., Lluch, J., 2010. Learning words using augmented reality, in: Advanced Learning Technologies (ICALT), 2010 IEEE 10th International Conference On. IEEE, pp. 422–426.

Kato, H., Billinghurst, M., 1999. Marker tracking and hmd calibration for a video-based augmented reality conferencing system, in: Proceedings 2nd IEEE and ACM International Workshop on Augmented Reality (IWAR'99). IEEE, pp. 85–94.

Kaufmann, H., 2003. Collaborative augmented reality in education. Institute of Software Technology and Interactive Systems, Vienna University of Technology.

Klopfer, E., Perry, J., Squire, K., Jan, M.-F., 2005. Collaborative Learning Through Augmented Reality Role Playing, in: Proceedings of Th 2005 Conference on Computer Support for Collaborative Learning: Learning 2005: The Next 10 Years!, CSCL '05. International Society of the Learning Sciences, Taipei, Taiwan, pp. 311–315.

Lin, K.-C., Wang, S.-C., 2012. Situated learning for computer fabrication based on augmented reality. Lecture Notes in Information Technology 23, 249.

Mahadzir, N.N., Phung, L.F., 2013. The use of augmented reality pop-up book to increase motivation in English language learning for national primary school. Journal of Research & Method in Education 1, 26–38.

Marshall, P., 2007. Do tangible interfaces enhance learning?, in: Proceedings of the 1st International Conference on Tangible and Embedded Interaction. ACM, Baton Rouge, Louisiana, pp. 163–170.

Masmuzidin, M.Z., Aziz, N.A.A., 2018. The Current Trends of Augmented Reality in Early Childhood Education. The International Journal of Multimedia & Its Applications (IJMA) 10, 47.

McCulloch, W.S., Pitts, W., 1943. A logical calculus of the ideas immanent in nervous activity. The bulletin of mathematical biophysics 5, 115–133.

Munsinger, B., Quarles, J., 2019. Augmented Reality for Children in a Confirmation Task: Time, Fatigue, and Usability, in: 25th ACM Symposium on Virtual Reality Software and Technology. ACM, p. 36.

Newman, W.M., 1968. A system for interactive graphical programming, in: Proceedings of the April 30--May 2, 1968, Spring Joint Computer Conference. ACM, pp. 47–54.

O'Brien, H.L., Toms, E.G., 2005. Engagement as process in computer-mediated environments, in: Poster Presented at the Annual Conference of American Society for Information Science and Technology, Charlotte, NC.

Olson, J.S., Kellogg, W.A., 2014. Ways of Knowing in HCI. Springer.

Ortiz, A., Vitery, C., González, C., Tobar-Muñoz, H., 2018. Evaluation of an Augmented Reality Multiplayer Learning Game, in: Göbel, S., Garcia-Agundez, A., Tregel, T., Ma, M., Baalsrud Hauge, J., Oliveira, M., Marsh, T., Caserman, P. (Eds.), Serious Games. Springer International Publishing, pp. 91–100.

Otsu, N., 1979. A threshold selection method from gray-level histograms. IEEE transactions on systems, man, and cybernetics 9, 62–66.

Radu, I., MacIntyre, B., Lourenco, S., 2016. Comparing Children's Crosshair and Finger Interactions in Handheld Augmented Reality: Relationships Between Usability and Child Development, in: Proceedings of the The 15th International Conference on Interaction Design and Children. ACM, pp. 288–298.

Ramus, F., 2003. Developmental dyslexia: specific phonological deficit or general sensorimotor dysfunction? Current opinion in neurobiology 13, 212–218.

Regenbrecht, H.T., Wagner, M., Baratoff, G., 2002. MagicMeeting: A Collaborative Tangible Augmented Reality System. Virtual Reality 6, 151–166. https://doi.org/10.1007/s100550200016

Santos, M.E.C., Polvi, J., Taketomi, T., Yamamoto, G., Sandor, C., Kato, H., 2015. Toward standard usability questionnaires for handheld augmented reality. IEEE computer graphics and applications 35, 66–75.

Sayed, N.E., Zayed, H.H., Sharawy, M.I., 2011. ARSC: Augmented reality student card an augmented reality solution for the education field. Computers & Education 56, 1045–1061.

Shin, J.G., Ng, G., Saakes, D., 2018. Couples Designing Their Living Room Together: A Study with Collaborative Handheld Augmented Reality, in: Proceedings of the 9th

Augmented Human International Conference, AH ’18. ACM, New York, NY, USA, pp. 3:1–3:9. https://doi.org/10.1145/3174910.3174930

Shoemaker, G., Tang, A., Booth, K.S., 2007. Shadow reaching: a new perspective on interaction for large displays, in: Proceedings of the 20th Annual ACM Symposium on User Interface Software and Technology. ACM, pp. 53–56.

Silva, M., Roberto, R., Teichrieb, V., 2013. Evaluating an Educational System Based on Projective Augmented Reality. https://doi.org/10.5753/CBIE.SBIE.2013.214

Singhal, S., Bagga, S., Goyal, P., Saxena, V., 2012. Augmented chemistry: Interactive education system. International Journal of Computer Applications 49.

Treiman, R., 2017. Learning to Spell Words: Findings, Theories, and Issues. Scientific Studies of Reading 21, 265–276. https://doi.org/10.1080/10888438.2017.1296449

Vaughan-Nichols, S.J., 2009. Augmented reality: No longer a novelty? Computer 42, 19–22.

Wise, A.F., Antle, A.N., Warren, J., 2017. Explanation-giving in a collaborative tangible tabletop game: Initiation, positionality, valence and action-orientation.

Wise, A.F., Antle, A.N., Warren, J.L., May, A., Fan, M., Macaranas, A., 2015. What Kind of World Do You Want to Live In? Positive Interdependence and Collaborative Processes in the Tangible Tabletop Land-Use Planning Game Youtopia., in: CSCL.

Wojciechowski, R., Cellary, W., 2013. Evaluation of learners’ attitude toward learning in ARIES augmented reality environments. Computers & Education 68, 570–585.

Wojciechowski, R., Walczak, K., White, M., Cellary, W., 2004. Building virtual and augmented reality museum exhibitions, in: Proceedings of the Ninth International Conference on 3D Web Technology. ACM, pp. 135–144.

Wu, H.-K., Lee, S.W.-Y., Chang, H.-Y., Liang, J.-C., 2013. Current status, opportunities and challenges of augmented reality in education. Computers & education 62, 41–49.

Yuen, S.C.-Y., Yaoyuneyong, G., Johnson, E., 2011. Augmented reality: An overview and five directions for AR in education. Journal of Educational Technology Development and Exchange (JETDE) 4, 11.