

AUGMENTED THREE-DIMENSIONAL PRINTED
MODELS FOR STUDENTS WITH VISUAL
IMPAIRMENTS

A Dissertation

Presented to the Faculty of the Graduate School

of Cornell University

in Partial Fulfillment of the Requirements for the Degree of

Doctor of Philosophy

by

Lei Shi

December 2019

© 2019 Lei Shi
ALL RIGHTS RESERVED

AUGMENTED THREE-DIMENSIONAL PRINTED MODELS FOR STUDENTS WITH VISUAL IMPAIRMENTS

Lei Shi, Ph.D.

Cornell University 2019

Three-dimensional models are important learning tools for students with visual impairments (VIs). These models are especially useful in presenting abstract concepts like the structure of molecules. Recently, teachers and educational institutions have started using 3D printers to create models for students with VIs. However, unlike diagrams that are marked with text labels and different colors, a printed model lacks important auxiliary information that can help students understand the model.

To address this challenge, we augmented printed models with interactivity, and created a new learning genre for students with VIs, augmented 3D printed models (A3Ms). A3Ms sense students' exploration behavior and provide multi-modal feedback to help students understand a model. Unlike interactive educational technologies that have been widely adopted for a mainstream population, A3Ms are a new medium designed for students with VIs and have many unresolved questions: First, what sensing technologies will enable printed models to sense users' exploration behavior? Second, what interaction design is effective on A3Ms? Third, what design tools would support people to easily create A3Ms? This dissertation contributes three components that address these research questions.

The first contribution is the sensing technologies for A3Ms. Most printed models are made of passive materials (e.g., plastic), and do not have embedded

sensors. What technologies can enable a passive model to sense inputs from users? To answer this question, we explored two sensing technologies: acoustic sensing and computer vision. These technologies can be powered by sensors from mobile devices and can be easily deployed in classrooms at minimal cost.

The second contribution is a series of studies that explored the design space and design guidelines of A3Ms. First, we conducted an elicitation study with people with VIs to solicit user-defined input techniques. In addition, we hosted workshops with Teachers of the Visually Impaired (TVIs) to explore how to design A3Ms as effective teaching aids. These studies introduced new input and output techniques and helped us shape the design of A3Ms, which was further examined in a co-design study.

The third contribution is the design tools for A3Ms. To make A3Ms available in practice, we need to empower non-experts (e.g., TVIs) to create A3Ms easily. Initially, we built a graphical user interface-based design tool. Although sighted people can use the tool to create A3Ms, the interface was not accessible to designers with VIs. we further designed a tangible model design tool to address the accessibility issues.

At the end of this dissertation, we discuss future research directions of A3Ms. This dissertation will also benefit future research in accessible educational technologies

BIOGRAPHICAL SKETCH

Lei Shi's research lies at the intersection of human-computer interaction and accessibility. In his research, he follows user-centered design methods to build intelligent systems to address accessibility challenges. His work frequently appears at the top-tier human-computer interaction conferences including CHI, UIST, ASSETS, and MobileHCI. He won first place at the Student Research Competition at ASSETS. In addition to his work at Cornell, he also collaborates with industrial researchers and has worked at IBM Research, Microsoft Research, and Facebook. Prior to Cornell, he received double Bachelor's Degrees in Electrical Engineering and Industrial Design from Zhejiang University. His Ph.D. study was supported by an Oath Fellowship and the National Science Foundation.

To Youth and Growth

ACKNOWLEDGEMENTS

This dissertation would not have been possible without the support of my advisor, committee members, friends, and family.

Firstly, I would like to express my deep gratitude to my advisor, Shiri Azenkot, who has been guiding and supporting me from the start of my Ph.D. study. As an advisor, she taught me how to conduct rigorous research to understand and address accessibility challenges. Her passion for technologies and pursuit of knowledge inspired me deeply. Her philosophy and scholarship about design and technology underpinned my research and shaped every chapter of this dissertation.

I would also like to thank the members of my committee who provided indispensable perspectives for my research work. Serge Belongie and his students helped me design the first computer vision-based prototype in Chapter 4. François Guimbreti re spent valuable time with me to discuss potential research ideas, which led to the tangible model design tool in Chapter 6.

Further, I would like to thank the generous financial support from many institutions. My research was supported in part by Verizon Media Group and the National Science Foundation under grant # 1746123.

Many people helped me during my doctoral work. This includes my colleagues at Cornell Information Science and Cornell Tech, mates in the Enhancing Ability Lab and Connected Experience Lab, friends from my undergraduate program, and Wenzhounese who shared their New York experiences with me.

Lastly, I extend my huge gratitude to my parents who always hold strong belief in me and provide unconditional support. My mother taught me the importance of humanity and care. My father encouraged me to be curious about everything and focus on impact. It is my great fortune to be your son.

CONTENTS

Biographical Sketch	iii
Dedication	iv
Acknowledgements	v
Contents	vi
List of Tables	viii
List of Figures	ix
1 Introduction	1
1.1 Sensing Technologies	4
1.2 Design Studies	5
1.3 Design Tools	7
1.4 Content Organization	9
2 Related Work	11
2.1 The History of Tactile Learning Materials	11
2.2 Accessible and Printed Tactile Learning materials	13
2.3 Augmenting 3D Printed Materials with Interactivity	14
2.4 Design Tools for Tactile Materials	18
2.5 Summary	21
3 Design Process	24
3.1 The Vision and Design Goals	24
3.2 User-Centered Design	27
3.3 Timeline and Relations	29
4 Sensing Technologies	31
4.1 Tickers and Talker: An Acoustic Sensing Approach	32
4.1.1 Introduction	32
4.1.2 Design Exploration and Formative Study	33
4.1.3 The Design of Tickers	37
4.1.4 The Design of Talker	44
4.1.5 Evaluation	47
4.1.6 Discussion	55
4.2 Talkit: A Computer Vision Approach	59
4.2.1 Introduction	59
4.2.2 Exploratory Study	60
4.2.3 Talkit	62
4.2.4 Evaluation	67
4.3 Summary	77

5	Design Studies	79
5.1	The Elicitation Study	80
5.1.1	Introduction	80
5.1.2	Study Design	82
5.1.3	Findings: Exploration	89
5.1.4	Findings: Elicitation	98
5.1.5	Proposed Design	101
5.1.6	Discussion	104
5.2	The Workshop Study	107
5.2.1	Introduction	107
5.2.2	Study Design	108
5.2.3	Findings	111
5.2.4	Discussion	118
5.3	The Co-Design Study	121
5.3.1	Introduction	121
5.3.2	Talkit++	122
5.3.3	Study Design	126
5.3.4	Design Deliverables	130
5.3.5	Findings	136
5.3.6	Discussion	139
5.4	Summary and Design Guidelines	142
6	Design Tools	145
6.1	Markit: A GUI-Based Design Tool	146
6.1.1	Introduction	146
6.1.2	The Design and Implementation of Markit	147
6.1.3	Evaluation	150
6.1.4	Results	153
6.1.5	Discussion	156
6.2	Molder: A TUI-Based Design Tool	158
6.2.1	Introduction	158
6.2.2	Design Requirements	161
6.2.3	The Design of Molder	165
6.2.4	Evaluation	179
6.2.5	Discussion	188
6.3	Summary	191
7	Discussion and Future Work	192
7.1	The Design of Tactile Features	192
7.2	The Fabrication of Tactile Learning Materials	194
7.3	Sensing Technologies	195
7.4	The Design of Interactive Elements	195
8	Conclusion	197

LIST OF TABLES

4.1	The ten statements (S1 S10) of the modified Standard Usability Scale (SUS) of Talkit’s evaluation study, with their histograms and means scores. Each statement was scored from 1 (stands for strongly disagree) to 7 (stands for strongly agree). For positive statements (S1, S3, S5, S7, S9), higher scores are better. For negative statements (S2, S4, S6, S8, S10), lower scores are better.	74
5.1	The four identified hand postures.	93
5.2	The eight gestures identified in the Exploration section.	96
5.3	The Gesture-Activity matrix. We marked all the combinations of exploration activities and gestures that appeared in the Exploration section of the study.	97
5.4	The total and distinct Techniques broken down by different interaction modalities.	98
5.5	Six referents and their max-consensus (MC) values, consensus-distinct ratios (CDRs), and distinct input actions that were above a consensus threshold of two. The MC and CDRs were also broken down into three input modalities. Two referents didn’t get any speech interaction. For each distinct input action, we list the number of the participants reporting in the “#” column.	102
5.6	The functions and interactions for the proposed design.	103
5.7	Different groups and their chosen models in the workshop study	110
6.1	The ten statements (S1 S10) of the modified Standard Usability Scale (SUS) of Markit’s evaluation study, with their histograms and means scores. Each statement was scored from 1 (stands for strongly disagree) to 7 (stands for strongly agree). For positive statements (S1, S3, S7, S9), higher scores are better. For negative statements (S2, S4, S5, S6, S8, S10), lower scores are better.	155
6.2	The ten statements (S1 S10) of the modified Standard Usability Scale (SUS) of Molder’s evaluation study, with their histograms and means scores. Each statement was scored from 1 (stands for strongly disagree) to 7 (stands for strongly agree). For positive statements (S1, S3, S5, S7, S9), higher scores are better. For negative statements (S2, S4, S6, S8, S10), lower scores are better.	187

LIST OF FIGURES

3.1	The vision of A3Ms. We think designers should be able to create A3Ms easily. The created A3Ms will be uploaded to an online community, where people share their models. Students with VIs can download and print A3Ms from the online community and start learning the models with interactive technologies.	25
3.2	The timeline of the projects presented in this dissertation. The projects are labelled with the names of corresponding chapters and sections.	30
4.1	A user strums a Ticker.	33
4.2	The first prototype that uses acoustic sensing. The prototype is designed as a molecule model and has three markers showing label locations and a scaffold supporting a mobile phone that speaks the labels when the markers are tapped.	34
4.3	Three prototypes that uses acoustic sensing. (a) The first prototype required a fixed mobile phone propped in the scaffold. (b) The second prototype included a case for a mobile phone. (c) The third prototype didn't require a fixed mobile phone; it could be placed on a desk as the user held the model.	39
4.4	Ten markers with a quarter for scale. On the top left is a Ticker. .	40
4.5	The structure of a Ticker.	41
4.6	Ten markers with a quarter for scale. On the top left is a Ticker. .	41
4.7	An H ₂ O molecule with three atoms labelled with Tickers.	43
4.8	A matrix showing the similarity between Ticker tones in a molecule model. The similarity is measured by the Euclidean distance between Ticker tones' amplitude spectral densities. Lower values means more similar. Tones from different Ticker types and locations are relatively distinct, which is promising. For examples, pairwise comparison of A-O to J-O (i.e., different Ticker types) has a high value. Pairwise comparisons for J-O, J-H ₁ and J-H ₂ (i.e., different locations) have high values. In an ideal situation, the center diagonal would be dark blue, and the rest of the matrix would be dark red.	45
4.9	Four printed models labeled with Tickers.	48
4.10	The mean accuracy for each participant and each model. Higher values are better.	53
4.11	The mean accuracy of the three models and misclassification rates for each of their Tickers. Nine participants activated each Ticker, producing a total of 72 data points. Tickers on the Molecule and Animals models were confused more frequently than Tickers on the Terrain model.	54

4.12	Mean scores for subjective statements. 1 stands for strongly disagree and 7 stands for strongly agree. Error bars represent 1 standard deviation. For positive statements (blue; S1, S2, S4, S8), higher values are better. For negative statements (yellow; S3, S5, S6, S7, S9), lower values are better.	55
4.13	A user with VIs interacts with a globe model using Talkit. She sticks a simple sticker (b) on her fingernail and slides a tracker (c) on the printed scaffold. She then launches the Talkit application (a) on her computer and interactively explores the model with its audio annotations.	60
4.14	Two required accessories for Talkit.	64
4.15	A user points her finger on the edge of the purple hotspot. The Talkit application identifies the center point of the sticker (marked in yellow dot) as the position of the finger and finds faces (outlined in green) that overlap with a circle (outlined in blue) around the center point.	66
4.16	Four models used in the evaluation study of Talkit. The models and their elements are highlighted in color, and their names and dimensions are listed in the figure as well.	69
4.17	The time participants spent finding all five elements on each model using Talkit. Mean times for all participants are shown in red lines.	72
4.18	A histogram of the time participants took to locate an element, categorized and colored by models.	72
5.1	The eight gestures identified in the Elicitation section of the study. The movements of dynamic gestures are marked with arrows. We highlight the hands that were performing the gestures and mark the remaining parts in grey.	82
5.2	Three printed models used in the study. From left to right: the Cell model, the Globe model, and the Map model.	84
5.3	A researcher was clustering the note cards. Another participating researcher took the photo.	88
5.4	Four identified hand postures from the Exploration section.	92
5.5	Four identified single-finger gestures from the Exploration section.	95
5.6	Four identified multi-finger gestures from the Exploration section.	95
5.7	A use case for the proposed design. A user selects an element with her index finger and says “more information” to retrieve further information about the element.	103
5.8	The seven printed models we used in the workshop study.	109
5.9	The participants used stickers to indicate where they want to add annotations. The top of the elephant model has some printing imperfections.	112

5.10	The participants modified the skull model. They used Play-Doh to indicate different tactile patterns on four components.	113
5.11	(a) Participants tried existing A3Ms with Talkit application running on a Macbook. The application uses the webcam of the Macbook. (b) Unsatisfied with the orientation settings, a participant rotated the webcam to an egocentric way.	119
5.12	A visually impaired student is using a mobile application to learn the interactive Plane model. The application speaks audio information when the student explores the model and highlights different components on the model with colors for low vision students. The iPad is on a stand so that the camera captures the student's interactions with the model.	122
5.13	The user interface of the Talkit++ application, which is accessible with the built-in VoiceOver screen reader.	124
5.14	Two ways a user could position an iPad using a stand.	125
5.15	Different designs of the Volcano model. The final design includes two 2D trackers, which are highlighted in red.	131
5.16	The Plane model designed for T2 and her students.	132
5.17	The materials and final design of the Map model.	132
5.18	The interactive elements on the Volcano model.	133
5.19	The animation of the Volcano model.	133
5.20	The interactive elements on the Plane model.	134
5.21	The interactive elements on the Map model.	134
6.1	A designer downloads a model of a globe (b) and imports it into Markit. She adds a tracker scaffold (c) and augments areas on the model with textual information (a).	147
6.2	For an input model (a), a user uses Markit to add a tracker scaffold. Markit outputs a printable model in STL format (b) and decimates the model (c) for future edits. Then, the user marks the model in colors and adds annotations on it (d). Finally, Markit publishes the model's annotation data (e).	148
6.3	The Markit user interface. On the left is the Markit toolbar, and on the right is a perspective view of a working model. A user has selected several faces, which are highlighted by Blender in yellow, and will mark them with the information in the Demarcation section.	149
6.4	Four models used in the evaluation study of Markit. The four models are same as the ones in Figure 4.16. The models and their elements are highlighted in color, and their names and dimensions are listed in the figure as well.	152
6.5	The mean time participants spent on each model, broken down by the different sections in Markit.	154

6.6	A designer is using Molder to modify a map model. (a) He prints a draft map model and places it in the Physical Ruler. (b) Then, the designer uses his finger to select buildings on the printed model and choose desired functions. (c) When he finishes, he prints out the final model, which has a variety of tactile patterns and a bigger size.	160
6.7	The user interface of the map creation web page.	166
6.8	The design of the Tangible Widgets.	167
6.9	The Physical Ruler and the Indicator. There are three markers and one stopper on the Physical Ruler. The Indicator has two stoppers and a rigid tactile pattern on its right side.	169
6.10	Three tactile patterns on a same building. From left to right, the added tactile patterns are Icon, Line, and Dot. The building is marked in yellow and the added patterns are marked in grey. . .	170
6.11	To add a braille label, a designer selects a building and the Braille function, and (a) enter texts on a popup textbox. (b) Molder processes the command and update the model. The braille label entered here is "Hotel."	171
6.12	A designer can use the Indicator to change the size of a model. The red lines in the images stand for the right end of the Indicator. The blue rectangles represent the final sizes of the model. . .	172
6.13	The user interface of the Molder Server. A user can select a building in the selection window and perform Molder functions. The model will be updated in real time.	173
6.14	Molder automatically creates a draft model with the geographic information provided from a designer. (a) First, Molder imports 2D building outlines. (b) Then, it combines overlapping units to a single building (e.g., the aqua building has four units) and assigns high-contrast colors and IDs to all buildings. (c) Molder also adds a base (marked in grey) and a small cube (marked in red) on the left top corner to indicate the orientation of the model. (d) Last, Molder extrudes the 2D building outlines to 3D meshes.	175
6.15	From left to right: a shallow draft model, an unedited regular draft model, and a final model.	176
6.16	Three draft models we used in the study. From left to right: Model A, Model B, and Model C.	181
6.17	The nine tasks in the study, including two braille labels (T2, T3), one audio label (T5), four tactile patterns (T1, T4, T6, and T7), one deletion (T8), and one scale task (T9).	183
6.18	The time participants spent on each task. T2, T3, and T5 required text entry. T1, T4, T6, and T7 are related to different tactile patterns. T8 is a deletion task and T9 is a scaling task.	184
6.19	The count of the modeling activities in the freelance test.	186

CHAPTER 1

INTRODUCTION

Evolving technologies have shaped the ways that we learn information. Yet, without accessible design, the technologies also lead to the inequalities, biases, and discrimination between people with varied abilities. For example, while sighted people can readily enjoy interactive online courses, students with visual impairments¹ (VIs) still face challenges in understanding basic visual content. This dissertation explores how to provide effective and equal information access in education. Specifically, our research aims to help students with VIs to learn from visualizations better.

Students with VIs have challenges to understand visualizations, which are widely used in instructional materials. Visualizations (e.g., images and diagrams) are widely used in education to represent information that cannot be conveyed by text alone. To receive an equal education to their sighted peers, students with VIs use non-visual tactile materials to access these visualizations. Tactile graphics are one of the most commonly used tactile materials. These graphics are raised line drawings produced by trained professionals using specialized equipment (e.g., thermoform, braille printers). In addition to tactile graphics, tactile models can be used to communicate a variety of visuals found in textbooks. Models can convey information more effectively than tactile graphics (Kumar et al., 2001), and studies have shown that students who studied with models tended to perform better on exams (Rule et al., 2011). Models are particularly beneficial when original referent objects are too big, small,

¹In this dissertation, we will use “visually impaired” and “people with visual impairments” to describe people that have different levels of visual impairments, including low-vision people and people who are totally blind.

fragile, or dangerous to be presented in the classroom. For example, a DNA molecule is very small and the structure of a DNA molecule is too complex to describe in texts to students with VIs, so teachers may supplement a science curriculum by introducing a tactile model to convey DNA's double helix structure.

Since 3D models are such important educational resources for students with VIs, teachers have used various expensive and laborious techniques to acquire or create them. When possible, some teachers purchased the limited types of commercial tactile models that were designed specifically for students with VIs. As with any product for a small market, only a small number of models were available, and they were usually expensive. Teachers often resorted to creating their own tactile representations using arts and crafts materials (Rule et al., 2011; Teske et al., 2014). These hand-made representations can take a lot of time and energy to create, and, while customized to the student and the lesson, their fidelity was limited to the teacher's skills and resources.

Recent developments in consumer-grade 3D printers and the thriving maker culture present exciting opportunities to transform access to tactile materials. 3D printers can not only empower teachers to produce versatile tactile models (Buehler et al., 2014), but they can also enable students with VIs to generate their own tactile visualizations (Kane and Bigham, 2014). The maker movement has spawned a global community of hobbyists, who are eager to share resources and solve socially-motivated problems (Hatch, 2013); thousands, if not millions of "makers," around the world create and share 3D models and accessibility solutions. For example, makers, who are hobbyists, amateurs, and friends of people with VIs, shared models designed to help people with VIs learn concepts on Thingiverse (Buehler et al., 2015).

Nevertheless, the kind of information that can be effectively represented with 3D printed models is limited. A commercial globe model for sighted people can provide a rich amount of information through text labels and different colors, while 3D printed models are usually monochrome without printed labels. Although some model designers have added braille labels on 3D models, only a small minority of people with VIs read braille (National Federation of the Blind, Jernigan Institute, 2009). Moreover, braille requires a lot of space and can easily be confused with other tactile features in a model. Thus, students with VIs, especially those who do not know braille, can have trouble understanding these models, and must rely on additional assistance from other people.

Facing these opportunities and challenges, we augmented printed models with interactivity, and created a new learning genre, augmented 3D printed models (A3Ms). When a student explores an A3M, the model senses the student's exploration behavior, and provides multimodal feedback to help the student learn concepts. We envision that in the future, 3D printers will be highly available, and students can print models at home and interact with them directly. On the other hand, teachers will use appropriate tools to design A3Ms as part of their course preparation. To achieve this vision, A3Ms should be easy to access, easy to use, and easy to design.

This ambitious vision leads to a series of questions. For example, most printed models are made of plastic and do not have sensing capabilities, how can a passive printed model provide interactive feedback? Moreover, unlike long established educational technologies that have been widely adopted, A3Ms are a new medium designed for students with VIs, how should we design A3Ms as effective learning tools? Lastly, how should we enable teachers

and friends of students with VIs, who are non-expert designers, to easily design A3Ms? To address these difficulties, we derive the following research questions:

1. What sensing technologies will enable printed models to sense users' exploration behavior?
2. What interaction design is effective on A3Ms?
3. What design tools would support people to easily create A3Ms?

This dissertation answers these questions by developing sensing technologies, exploring design space and guidelines of A3Ms, and building design tools to create A3Ms.

1.1 Sensing Technologies

Since most printed models are made of passive materials like plastic, how can a model detect where the student is touching? Adding embedded sensors and electronic components can make models interactive. However, this method requires additional expenses and specific skills (e.g., assembly and maintenance) that most non-experts (e.g., teachers) do not have. We need to augment models with sensing technologies that do not rely on ad-hoc electronic components or specialized equipment. To augment models with sensing capabilities, we explored two sensing technologies that do not rely on ad-hoc electronic components or specialized equipment.

First, we designed Tickers and Talker (Shi et al., 2016b), which add interactivity on models using acoustic sensing. Tickers are printable percussive components, which can generate unique sounds when flicked. Talker is a signal

processing application that recognizes the Ticker sounds using machine learning algorithms and speaks audio labels. Talker only requires the audio signals from a mobile device to detect a user's exploration behavior. While Tickers and Talker received positive feedback from users, they are sensitive to noise and may not be suitable for daily use in noisy classrooms.

To find a more robust approach, we designed Talkit (Shi et al., 2017b), a computer vision-based system, that detects the model and finger gestures using simple accessories (i.e., paper stickers and fiducial tags, adopted from Chilitags (Bonnard et al., 2013)), and an RGB camera on a computing device. The system tracks the model with fiducial tags and identifies the user's finger with the attached sticker. Based on a model's position and a user's input, Talkit provides feedback to help the user learn with the model. Compared to Tickers and Talker, Talkit is robust to environment noise.

1.2 Design Studies

Unlike designing interactions for traditional media (e.g., audiobooks), interaction design for accessible 3D printed models is not established and we do not yet know what design is effective. While there are existing products and prior work on interactive educational models (e.g., Tiptoi[®] (2019)), most of these models were designed for sighted people. In the field of accessibility, despite the emerging research on sensing technologies for A3Ms, little research has explored the design perspective of A3Ms. Specifically, how should users with VIs retrieve interactive information from A3Ms? What interactive elements should we add on A3Ms for people with VIs?

To answer these questions, we conducted three studies. First, we conducted an elicitation study with 12 participants with VIs to explore different input techniques (Shi et al., 2017a). The study aimed to explore the design space of A3Ms with people with VIs. In a typical elicitation study, researchers prompt a user with the effects of an action, and the user is expected to provide the causes of the action (Good et al., 1984). In our study, we elicited user-defined input techniques that would trigger various functions from an A3M. In addition to the typical elicitation procedure, we also observed participants' gestures and exploration strategies as they explored tactile models that are not interactive. These observations helped us understand the exploration behavior of people with VIs and contribute to the fundamentals of A3Ms. The study provided design recommendations for interactions as well as design implications for future A3Ms. However, this study mostly focused on input techniques, and solicited feedback only from adults with VIs.

To include feedback from other stakeholders and further investigate how would education practitioners design A3Ms as effective teaching aids, we led two design workshops with 35 Teachers of the visually impaired (TVIs) (Shi et al., 2019). TVIs are domain experts trained to teach students with VIs. The TVIs modified 3D models that researchers prepared and proposed interactive elements they wanted to add on the models. Through the workshops, the TVIs provided suggestions about model modification (e.g., the size of physical models should be bigger), interactive elements (e.g., adding visual animations for low vision students), interaction techniques (e.g., supporting speech input), and pedagogical considerations (e.g., providing an overview before details).

The findings from the elicitation study and the workshops helped us refine

the design of A3Ms, which were further examined in a co-design study. With the feedback from the previous studies, we built Talkit++ (Shi et al., 2018), a mobile application that uses computer vision to track finger gestures and model positions. The application provides auditory information, renders visual feedback, and supports simple speech commands. Compared to Talkit, which is a desktop application, Talkit++ is easier to setup and can support more interactive elements (e.g., visual feedback). To further understand how A3Ms and the Talkit++ application work in the field, we conducted a co-design study, where we collaborated with three TVIs individually to design and deploy sample A3Ms over seven weeks.

Together, these studies revealed ten design guidelines, which will enable future model designers (e.g., TVIs, brailleists) to create better A3Ms. These guidelines are categorized in three groups. For example, interactive models should (1) have effective tactile features (e.g., distinctive tactile patterns between components), (2) contain both auditory and visual content (e.g., explanatory animations), and (3) consider pedagogical methods (e.g., overview before details).

1.3 Design Tools

To make A3Ms more available, we need to enable non-expert designers (e.g., TVIs and friends of students with VIs) to design A3Ms. Currently, to design an A3M, a designer must use modeling software applications, which require professional training and rely on visual feedback. While skilled designers may be able to design A3Ms using different tools together, a lot of people who need to design A3Ms (e.g., TVIs) are not trained as professional designers and not ca-

pable of performing these tasks. In addition, people who would be interested in creating the models may be visually impaired. The complexity and inaccessibility of current modeling applications thus become major barriers when designing A3Ms.

To empower non-expert designers to design A3Ms, we developed two design tools. First, we created Markit (Shi et al., 2017b), which can be used to mark model elements and associate them with interactive annotations. Markit was implemented as an add-on to Blender (2019), a popular 3D modeling application among designers. Markit provides step-by-step guidance for creating augmented models for non-experts. After importing a downloaded model into Markit, a maker can modify the model and associate annotations with elements on the model. The annotations can be used in Talkit++. A user study showed that sighted participants learned how to use Markit quickly and added two annotations to a model in an average of 275 seconds (SD = 70). However, we also found that participants felt it was cumbersome to do 3D navigation and selection on a 2D graphical user interface. More importantly, the user interface is not accessible to people with VIs, and some teachers and designers are visually impaired.

To make the design process of A3Ms accessible, we designed Molder, a tangible design tool that enables users to design an A3M by physically manipulating a draft model. Using similar algorithms from Talkit and Talkit++, Molder detects the position of the model and the user's behavior. When designing a model, a user needs to print a draft model. Then, the user selects different components of the model by touching the components, and performs simple modifications like changing tactile patterns, sizes, and adding braille labels. For

example, a user points out a region on a map model, and she assigns a tactile pattern on the region. The system also allows the user to add interactive elements through speech input.

1.4 Content Organization

As explained above, we will introduce A3Ms as three components in this dissertation: sensing technologies, design studies, and design tools. However, these three components are highly connected and dependent on each other. As a matter of fact, each of the projects demonstrated in this dissertation has more or less touched all of these three components. In Chapter 3, we will explain the design goals and process of these projects, which will examine the projects from the perspective of user-centered design and help readers understand the rationale behind our design decisions.

The purpose of this dissertation is to introduce the three core components of A3Ms for future research. Although technology evolves, the lessons we learned from these components should be able to provide suggestions and guidance for future researchers and designers. For example, if we used another sensing technology other than computer vision, the gestures we chose in the elicitation study could be different. However, we will still need to create models with salient tactile features according to the design guidelines we found from multiple studies.

The rest of this dissertation is organized as follows: Chapter 2 reviews related work. Chapter 3 explains the overall design goals and process of the projects. Chapter 4 describes the technical details and evaluation of two sensing technologies. Chapter 5 introduces different studies we conducted to explore

the interaction design of A3Ms. These include the elicitation study, the workshop study, and the co-design study. In that chapter, we will also introduce Talkit++, the mobile application that builds on the findings of these studies. Chapter 6 presents the design tools: Markit and Molder. Finally, Chapter 7 discusses how these systems and studies can contribute to the field of accessible educational technologies. we will also point out future work for A3Ms. Chapter 8 concludes this dissertation with a summary of contributions.

The content of this dissertation was derived from previous publications. More specifically, the text of Chapter 2 was derived from the related work sections of four previous papers (Shi et al., 2016b, 2017b,a, 2019). Chapter 4 was derived from two papers at CHI '16 and UIST '17 (Shi et al., 2016b, 2017b) and two demonstration publications at ASSETS (Shi, 2015; Shi et al., 2016a). Two papers at ASSETS '17 and CHI '19 contributed to the content of Chapter 5 (Shi et al., 2017a, 2019). The text of Chapter 6 was derived from a prior paper (Shi et al., 2017b). Readers may refer to the original papers for more detailed information about individual projects.

CHAPTER 2

RELATED WORK

Augmented 3D printed models, as their names indicate, are printed tactile learning materials enhanced with interactivity. A3Ms build on the prior work from many fields, and we will review work from three most related topics. First, we will start with basic tactile learning materials, and we will introduce how 3D printing technologies increase the availability of tactile learning materials. Then, we will discuss how prior work augment 3D printed materials with interactivity, which involves different sensing technologies. The last section will focus on methods and tools to create tactile materials, which inspired the design tools in this dissertation. This chapter will help readers understand the background of this dissertation.

2.1 The History of Tactile Learning Materials

Tactile materials play an important role in the history of education. In the 19th century, Johann Heinrich Pestalozzi argued that students should learn through their senses and physical activity (Pestalozzi, 1803). Following this idea, education practitioners created tactile learning materials for their students. Friedrich Froebel, also known as the founder of kindergartens, developed a set of physical objects like balls, blocks, and sticks. These objects were designed to help children recognize the common patterns and basic forms in daily life (Lilley, 1967). Nowadays, we can find different variations of Froebel's physical objects throughout the world. For example, LEGO[®] blocks enable children to learn structures through physical manipulations. Maria Montessori extended Froebel's idea and encouraged children to use tactile objects to learn more con-

ceptual knowledge. Her idea also inspires modern tactile learning tools like Cuisenaire rods, which are used to teach numerical proportions (Zuckerman et al., 2005).

In addition to Froebel-inspired and Montessori-inspired tactile materials, which are mainly designed for young children, there are more advanced tactile learning materials available in modern classrooms and research communities. In a typical STEM lecture, you are likely to see a diverse collection of tactile materials like physical molecules models, terrain maps, and astronomy models, just to name a few. More recently, tactile learning materials were enhanced with interactivity. For example, Tiptoi[®] (2019) allows students to access audio annotations on tactile models using a pen. Students can potentially learn the tactile information of a model using a pen through proprioception. However, the experience of exploring a model using a pen is quite different from exploring the model using hands. Researchers also developed dynamic tangible interfaces that can present rich tactile information. Nakagaki et al. (2019) designed a system that enables people to understand the internal structure of 3D volumetric information through rendered dynamic shapes.

Prior work demonstrated the rich design space of tactile learning materials. However, most of those tactile materials, even those enhanced ones, were designed for sighted people. In addition, some of those materials and systems are expensive and difficult to maintain. This dissertation aims to explore a low-cost interactive tactile material for people with VIs.

2.2 Accessible and Printed Tactile Learning materials

Some traditional tactile learning materials are designed specifically for people with VIs. There are two major types of accessible tactile learning materials: tactile graphics and 3D models. Tactile graphics can be produced with expensive embossers that print raised dots and lines on special paper. Beside tactile graphics, few tactile models are available on the market (e.g., The DNA model and Globe model produced by the American Printing House). When commercial accessible visualizations are not available, TVIs often adopt Do-It-Yourself solutions. Even with simple handmade models, students with VIs can significantly improve their learning process (Rule et al., 2011).

The growing availability of 3D printers enables people to print tactile learning materials (Buehler et al., 2015, 2016; Hurst and Tobias, 2011). 3D printing technology is becoming more and more available. 3D printers have evolved from pieces of industrial equipment to personal tools, and are widely used in fabrication activities like prototyping and repairing (Lipson and Kurman, 2013). People can already print models with printers found in community maker spaces, schools, and libraries. As 3D printing technology continues to evolve, people with VIs will be able to print tactile learning materials at home, at school, or at their local library.

Researchers and practitioners have designed various printed tactile graphics and models for people with VIs. Grice et al. (2015) converted Hubble Space Telescope images into tactile graphics. Kolitsky (2014) 3D printed images from STEM disciplines like anatomy, along with embossed braille labels. Gual et al. (2011, 2012, 2014) explored different designs of printed tactile maps. For ex-

ample, they designed different tactile symbols as keys in maps. Most of these materials were designed by researchers as proofs of concept and evaluated with visually impaired adults. In this dissertation, we extended prior work by designing tactile materials with TVIs, as described in Section 5.2 and Section 5.3.

Among prior work, only a few projects involved students with VIs or teaching staff in their studies. McDonald et al. (2014) printed tactile aids to help students with VIs learn graphic design theory. Kane and Bigham (2014) taught students how to analyze and visualize Twitter data using Ruby programs, and used 3D printing technology to help students with VIs understand their visualizations. Stangl et al. (2014, 2015) designed printed tactile books with teachers and caregivers of students with VIs. Kim and Yeh (2015) further added movable components into printed tactile books and evaluated their designs with TVIs. Hu (2015) explored paradigms for 3D printed graphs with TVIs in a co-design process.

This growing number of accessibility research indicates the importance of printed learning materials to people with VIs. This dissertation explores how to augment and design these printed learning materials.

2.3 Augmenting 3D Printed Materials with Interactivity

Seeing the potential of 3D printed tactile materials, researchers have augmented 3D models with interactive elements to convey more information. Traditional printed tactile materials can only convey limited types of information, and there is not enough space for braille text (Brock et al., 2015; Holloway et al., 2018). Thus, researchers created augmented tactile materials that can sense a user's be-

havior and provide interactive feedback (e.g., descriptive audio labels) to help the user understand the materials. To sense a user's behavior on a printed material, prior work used different sensing technologies, which can be categorized into five types: embedding sensors into models, new fabrication techniques, combining conductive printing materials with touchscreens, acoustic sensing, and computer vision. In the following content, we will review prior projects by the types of sensing technologies they used. While most of the work described in this section aimed to address accessibility issues, we also include sensing technologies that added interactivity to 3D prints for general purposes (e.g., creating a functional printed prototype), as these sensing technologies can be potentially repurposed for accessibility usages.

Embedding sensors inside 3D printed objects is a common method to increase their interactivity. For example, Holloway et al. (2018) augmented 3D printed maps with interactive audio labels using an Arduino microcontroller and capacitive sensors. When a user touches the sensors, the microcontroller detects the signals and process corresponding feedback. Savage et al. (2013) used an embedded camera to turn hollow 3D printed models into interactive objects. Hook et al. (2014) demonstrated how to combine wireless accelerometers and 3D printed objects to rapidly prototype interaction designs. Several projects combined sensors with internal pipes. Willis et al. (2012) added optical pipes into 3D printed objects to make them capable of sensing gestures and displaying content. Savage et al. (2014) illustrated the design space of interactive printed objects that have internal pipes and embedded sensors, and they developed a tool for making interactive models with these pipes. In addition, Vázquez et al. (2015) added pneumatic elements to fabricate controls that could be used as both input and output devices. While these methods provide rich

interactive design opportunities with printed objects, they require additional electronics and assembly, which make it harder for TVIs and people with VIs to produce 3D models.

Researchers also explored new fabrication techniques to create interactive prints. Peng et al. (2015) built a 3D printer that prints soft objects with layered fabric. The printer can insert different sensors to add interactivity to the printed soft objects. In another work, Peng et al. (2016) designed a 3D printer that prints interactive electromechanical objects by automatically adding metal wire. The added wire forms coils, which induce magnetic fields when powered with electricity. Besides new printers, researchers also added new materials to fabricate interactive prints. For example, Ishiguro and Poupyrev (2014) used conductive materials to convert 3D printed objects into speakers, and created several interactive applications with the speakers. Zhang et al. (2017) fabricated conductive printed objects using a carbon-loaded filament and carbon spray. Then, they used electric field tomography to sense a user's input on the conductive printed objects. In addition to embedded sensors and new fabrication techniques, researchers have combined conductive materials with touchscreens to add interactivity to 3D printed materials. In prior work, researchers printed models with conductive parts and put them on a touchscreen (Götzelmann, 2016; Götzelmann, 2018; Schmitz et al., 2015; Taylor et al., 2016). When a user touches the conductive parts, the touchscreen recognizes the gesture and plays an audio label. Since the models must be in contact with a touchscreen, a user cannot explore the models freely in the air, and the designs of these models were usually limited to maps.

To explore a method that can augment prints with no embedded sensors or

special materials, prior work used acoustic sensing techniques. For example, Tejada et al. (2018) designed models with empty internal space. Each empty space is connected to an opening at the surface of a model. A user blows into the opening to produce a unique sound, which is picked up and recognized by a Python script to provide corresponding audio label. Similarly, Lamello (Savage et al., 2015b), Stane (Murray-Smith et al., 2008), and Acoustruments (Laput et al., 2015) also used acoustic sensing techniques to augment printed materials with interactivity. Tickers and Talker (Shi et al., 2016b), which will be described in Chapter 4, also adopt an acoustic sensing approach.

Computer vision has been applied to recognize users' exploration behavior on 3D models as well. Reichinger et al. (2016) used a depth sensor to allow people with VIs to explore tactile reliefs with audio guidance. They designed three gestures to retrieve audio labels, get detailed audio descriptions, and turn on or turn off audio feedback on tactile reliefs. Similarly, Shen et al. (2013) used a Kinect camera to identify users' exploration behavior on 3D objects. In these two projects, due to their configuration and algorithm design, a user with VIs needs to place models on a table to retrieve information, which prevents the user from freely exploring the models. Talkit (Shi et al., 2017b) and Talkit++ (Shi et al., 2018, 2019), which will be described in Chapter 4 and Chapter 5 respectively, also used computer vision to augment 3D printed tactile materials. Unlike prior work, Talkit and Talkit++ allow users with VIs to fully explore the 3D spatial information on models with a single RGB camera on a mobile device.

2.4 Design Tools for Tactile Materials

To design tactile materials, including those learning materials used by people with VIs, a designer needs to materialize his idea into 3D models using design tools. There are off-the-shelf design tools available. To further improve the experience of 3D modeling, researchers contributed new design tools for 3D modeling using tangible interaction techniques and augmented reality. In this section, we will start with the off-the-shelf design tools and the recent advances in 3D modeling. Most of these tools are designed for general purposes and are not specialized in designing tactile learning materials for people with VIs. To provide more context for this dissertation, we will review design tools for tactile maps and tactile graphics.

Most of the off-the-shelf design tools are designed for design professionals. For examples, the Fusion 360 software from Autodesk (2019a) and SolidWorks (2019) are used by industrial designers for computer-aided design and computer-aided engineering. The Maya software from Autodesk (2019b) is used for designing 3D animations and environments. SketchUp (2019) is popular among architects for designing 3D buildings. Blender (2019) is more generic and integrates several design functions from 3D modeling to animation design. While most of these tools are developed for professional designers, a few of tools are intended for amateurs. Tinkercad (2019) enables non-expert designers to create simple models using preset primitives. The Meshmixer software from AutoDesk (2019) allows makers to fine tune the designs of 3D models for better printing quality. All of these tools use graphical user interfaces (GUIs), and rely on visual feedback in the modeling process. In addition to GUI-based tools,

programming-based 3D designing tools are another promising way to create 3D models. For example, OpenSCAD (2019) is a widely adopted programming tool for model design. A designer can use a programming script provides functions to create primitives and manipulate them using geometric operations. Similarly, Yeh and Kim (2018) developed a web programming language for 3D modeling. However, these design tools are not specialized for tactile learning materials and may require several operations to perform a single task like adding braille.

Recently, researchers used tangible modeling techniques and augmented reality to make the modeling process more intuitive and easy to learn. Prior work allowed users to use physical objects as reference. For example, ModelCraft (Song et al., 2006) captures pen annotations and edits on physical 3D models. These captured annotations and edits are further used to edit original CAD models. Similarly, Makers' Marks (Savage et al., 2015a) allowed designers to attach stickers to a physical object to indicate their edits. Prior work also created modeling tools for sighted designers using augmented reality. MixFab (Weichel et al., 2014) is a mixed reality environment for personal fabrication, where a user can interact with a virtual model with gestures and introduce existing physical objects into the model design. Window-Shaping (Huo et al., 2017) is a mobile application that allows users to create digital 3D shapes directly on and around physical objects. RoMA (Peng et al., 2018) allows a designer to design and modify a 3D model digitally via an augmented reality headset and a controller while a robot arm is printing the model simultaneously. However, these 3D modeling tools largely relied on visual feedback, overlooking the needs of people with VIs.

In this dissertation, we focused on design tools for non-expert designers

to create tactile learning materials. While TVIs and Orientation and Mobility (O&M) specialists are responsible for creating tactile materials, studies showed that they were not trained as professional designers and faced challenges in designing and creating tactile materials for people with VIs (Giraud et al., 2017; Stangl et al., 2015; Rowell and Ungar, 2003; Sheppard and Aldrich, 2001). For example, Sheppard and Aldrich (2001) found that producing tactile materials for blind students was labor intensive for TVIs. Moreover, Stangl et al. (2015) found that non-expert designers needed assistance when design tactile books.

To address the challenges that non-expert designers had, researchers have designed technologies to lower the barrier of creating tactile models. This is critical for enabling non-experts, like TVIs, to create and print their own 3D models. Most researchers focused on how to automatically generate printable tactile materials from 2D images, especially 2D maps (Götzelmann and Pavkovic, 2014; Taylor et al., 2015). For example, TactileMaps.net (Taylor et al., 2015) is a web tool that allows visually impaired users to specify a location on an online map, and the tool automatically generates a corresponding tactile map model. In addition to maps, researchers also created tools for images in other STEM curricula. VizTouch software (Brown and Hurst, 2012) automatically generates tactile mathematical graphics from equations or csv files. Wedler et al. (2012) developed software that allows students with VIs to create and print chemistry models independently with minimal assistance from sighted co-workers. However, these techniques focused on computer generated tactile models, without enabling education experts (e.g., TVIs) to create and modify their own models, while education experts need to customize tactile materials to meet different needs of their students and clients.

Instead of using computer generated content, (Swaminathan et al., 2016) built a tactile display system that enables people with VIs to design their own 2D tactile graphics. The system used 3D printing techniques to create touchable raised lines on a board. The final model is a sculpted element attached to a solid background of the same material. Besides touching, the user can also interact with the system with pointing and speech interaction. However, these designed tactile graphics were fixed to the system and not suitable for distribution and daily uses.

However, most prior work focused on creating traditional tactile learning materials that do not have interactivity. Markit and Molder are novel design tools that enable designers to create A3Ms by adding interactive elements on models. Markit allows a maker to add audio labels on a 3D model. However, Markit is based on a graphical user interface, which relies heavily on visual feedback. Thus, TVIs and visually impaired users may have challenges using it. On the other hand, Molder incorporates tangible interaction, auditory feedback, and high-contrast visual feedback, and can be operated by designers with different visual abilities.

2.5 Summary

Prior work indicates the importance of printed tactile learning materials to people with VIs and motivates us to find appropriate sensing technologies for A3Ms. Despite the multiple applications of printed tactile learning materials, the kind of information that can be effectively represented with printed tactile learning materials is limited. Thus, researchers used different sensing technolo-

gies to make 3D printed materials more powerful. However, most of these sensing technologies require additional electronics, specialized hardware, and skills like wiring and soldering. Therefore, these sensing technologies may be difficult to implement in classrooms. In this dissertation, we added interactivity to printed tactile materials using acoustic sensing and computer vision, as described in Chapter 4. These two approaches are inspired from previous work and incorporate design implications we gained from people with VIs.

Besides sensing technologies, this dissertation further explores the design space of A3Ms through a series of studies. Prior work mostly focused on the technical challenges of A3Ms. The A3Ms in prior work only had straightforward interactions designed by researchers. Most of these models were not designed with TVIs, nor were they evaluated by students with VIs. This dissertation examined A3Ms with TVIs and students in Chapter 5. In addition, Chapter 5 also describes design guidelines for A3Ms.

To make A3Ms more available, we need to empower non-expert designers (e.g., TVIs) to create tactile learning materials. Most of the off-the-shelf design tools are designed for experts and not specialized for tactile learning materials. Although a trained designer can use them to create 3D models, most TVIs and O&M specialists, who are in charge of creating tactile learning materials for students with VIs, do not have the appropriate skills to use these tools. While prior research projects contributed easy-to-use design tools specialized for tactile learning materials, most of them focused on traditional tactile materials that do not have interactivity. In this dissertation, we contribute new design tools that are easy to use, and specialized for A3Ms with interactive elements.

In prior work, researchers also invented advanced design tools to make the

design process easier. They used tangible modeling techniques and augmented reality to connect digital models with real-world objects. These tools enable designers to use physical objects as references, and lower the barrier for model design. While these techniques can potentially be used for designing A3Ms, the design tools built in prior work heavily relied on visual feedback, which is not accessible to designers with VIs. In Chapter 6, we will introduce Molder, a design tool combining augmented reality and tangible modeling techniques. Molder builds on prior work and uses audio feedback and high-contrast visual feedback to make the modeling process accessible to designers with VIs.

CHAPTER 3

DESIGN PROCESS

This dissertation consists of several entangled projects we conducted over last four years. Throughout these projects, we also collected feedback from over one hundred participants. During this process, we evolved our research agenda based on the feedback we received from other researchers and our participants.

Despite the dynamic research agenda, we stuck to our vision and design goals by strategically following a user-centered design process. The massive feedback we collected provided design ideas for us. As researchers and designers, our responsibility is to make design decisions based on the feedback, instead of relying on feedback. In our projects, we designed systems to meet our goals. Then, we iterated and analyzed the designs through a user-centered design process.

In this chapter, we will start with our vision and design goals: what we want to achieve by conducting these research projects? Then, we will describe the user-centered design process we followed. Last, we will describe the timeline of the previous projects and the relations between them. While this dissertation categorizes the previous projects into three components (i.e., sensing technologies, design studies, and design tools), this chapter will help readers understand why we did these projects and the design decisions behind these projects.

3.1 The Vision and Design Goals

We aim to build powerful and accessible teaching aids for students with VIs. Nowadays, we have many teaching aids designed to improve learning expe-

rience. For example, slides are widely used in schools to help students better understand the content of lessons. Teachers have tools (e.g., Microsoft PowerPoint (2019)) to easily create slides, and students can download slides online to review lessons. However, most of these teaching aids are not accessible to students with VIs.

We think A3Ms, which combines interactive technologies and printed tactile features, will become the next-generation teaching aids for students with VIs. Figure 3.1 denotes our vision. Eventually, we hope to foster an online community that provides A3Ms to people with VIs. Designers (e.g., TVIs) will be able to easily create A3Ms and upload them to the online community, as similar to creating and uploading slides for sighted students. We envision that people with VIs will be able to access 3D printers in their daily lives and print A3Ms effortlessly from the online community. To support this ambitious vision, we

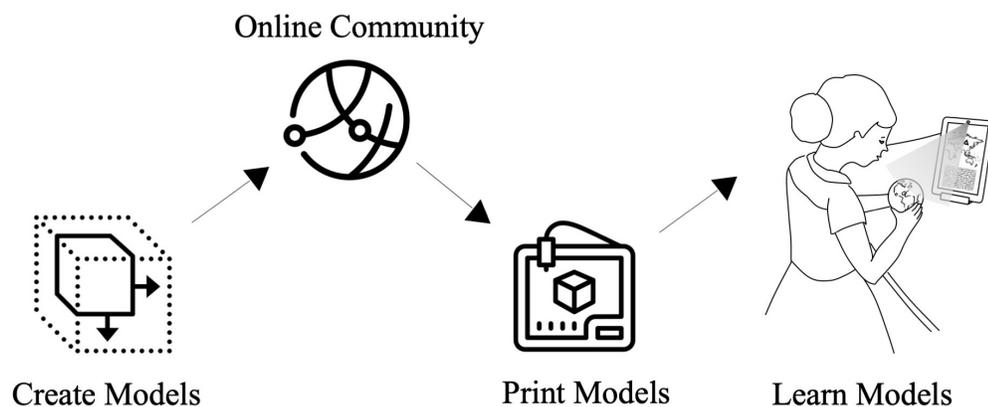


Figure 3.1: The vision of A3Ms. We think designers should be able to create A3Ms easily. The created A3Ms will be uploaded to an online community, where people share their models. Students with VIs can download and print A3Ms from the online community and start learning the models with interactive technologies.

have several design goals and challenges:

1. *A3Ms should be easy to access.*

Students with VIs should be able to access A3Ms with little added effort. To make A3Ms easier to access by students, these models should avoid complex assembly or expensive auxiliary components. However, most printed models are made of passive materials (e.g., Polylactic Acid and Acrylonitrile Butadiene Styrene). How can passive models sense and interact with students? To achieve this design goal, Chapter 4 discusses different sensing technologies that make passive models interactive .

2. *A3Ms should be easy to use.*

As teaching aids, A3Ms should contain meaningful content and their interactions should be intuitive. However, unlike existing educational technologies, A3Ms are a new media and we do not know yet the design guidelines for A3Ms. How should we design A3Ms as effective teaching aids? Chapter 5 describes several studies that help us refine the design of A3Ms.

3. *A3Ms should be easy to design.*

To make A3Ms more available, we need to support non-expert designers to create these models. The fewer efforts a designer has to spend to create an A3M, the more available these models will be. We envision that TVIs will design A3Ms. However, TVIs are not design professionals and do not have the skills to operate most 3D modeling applications. To achieve this goal, Chapter 6 demonstrates two easy-to-use design tools that enable non-expert designers to create A3Ms.

We used these design goals as guidance when making design decisions. For example, when we chose sensing technologies for A3Ms, we consider not only how many interactions they can afford but also think about how expensive they are. Thus, although we can create A3Ms by embedding advanced sensors and electronics inside prints, which can potentially support more gestures, we did not use them in the end because of their cost and availability.

3.2 User-Centered Design

The last section introduces our vision and design goals. To fulfill these design goals, we followed a user-centered design research method.

User-centered design is a broad term to describe the design processes that involve end-users and stakeholders (Abrams et al., 2004). The term gained popularity after the publication of a book entitled: *User-Centered System Design: New Perspectives on Human-Computer Interaction* (Norman et al., 1986). Then, human-computer interaction researchers enriched this concept by adding suggestions (Norman, 1988), principles (Shneiderman et al., 2016), and methods (Nielsen, 1994). In our projects, we involved stakeholders into different stages of the design process and chose varied design methods to refine the systems.

We used an iterative user-centered design process involving different prototypes and stakeholders. We designed prototypes, ran studies with stakeholders, and iterated on the designs of prototypes based on the results from the studies. The prototypes were designed with different levels of fidelity, ranging from Wizard-of-Oz prototypes to functional high-fidelity prototypes. In addition to different prototypes, we also involved different stakeholders during the

design process for varied design tasks. For example, we solicited feedback from TVIs when designing interactive content for A3Ms and tested these models with their students with VIs. As readers may find out later in this dissertation, we recruited people with VIs of different age groups in the studies, instead of focusing on young children and students. This is partially because of the difficulties of recruiting young students with VIs. In addition, we think A3Ms should benefit people of different ages and abilities.

At a high level, this dissertation consists of the iterative design processes of two relatively separate systems. The first system is designed to enable people with VIs to interact with 3D printed tactile materials (i.e., the first design goal). To design this system, we started with several prototypes using acoustic sensing and computer vision, which are described in Chapter 4. These are functional prototypes with limited features. Then, we ran studies to solicit feedback from different stakeholders, as described in Chapter 5. These prototypes and feedback contributed to the final system, Talkit++, which is used nowadays in classrooms. In Chapter 6, we detailed the design process of the second system, which aims to empower non-expert designers to design A3Ms (i.e., the second design goal). We started with a prototype based on Blender (2019). Then, we conducted several studies using the prototype, which exposed the limitations of graphical user interfaces and led to the final design.

At a lower level, each of the prototypes were developed in an iterative design process. For example, when designing the computer vision prototype, we first conducted a study using a Wizard-of-Oz prototype, which helped us refine the design. Then, we ran a usability study to test the refined prototype. Readers will find more details about the iterative design processes and study methods

in Chapter 4, Chapter 5, and Chapter 6.

3.3 Timeline and Relations

In this dissertation, we present our projects in three interconnected components. To help readers better understand the connections between each project, in this section, we will briefly describe the relations between these projects in chronological order.

In the Summer 2015, we started exploring the idea of A3Ms. We started with acoustic sensing and developed the first prototype (Shi, 2015). In the following Fall, we improved the prototype and designed Tickers and Talker (Shi et al., 2016b). We evaluated the prototype with people with VIs. The evaluation results indicated that the accuracy and bandwidth of acoustic sensing technology were limited, and adding more labels would inevitably decrease the accuracy of the acoustic recognition system. This process is documented in Section 4.1.

Considering the limited performance of the acoustic sensing approaches, we began investigating the potential of computer vision in Spring 2016. First, we conducted an elicitation study to solicit user-preferred input techniques, which is described in Section 5.1. The study provided a wide range of input techniques (e.g., gestures, speech input commands, and buttons), and we chose the ones that can be implemented using commodity devices to follow the first design goal. With the findings from the study, we built Talkit (Shi et al., 2016a, 2017b), the first computer vision-based prototype running on laptops, as described in Section 4.2. Moreover, to help people create A3Ms that can be used in Talkit, we built a GUI-based design tool, Markit (Shi et al., 2017b), as shown in Section 4.2.

From Fall 2017 to Summer 2018, we designed A3Ms and iterated the design of Talkit with TVIs and their students. First, we hosted two workshops with TVIs to understand how they would design and use A3Ms in their lessons. These workshops provided not only the design guidelines for A3Ms, but also the suggestions for improving Talkit, as described in Section 5.2. To address the limitations of Talkit, we developed Talkit++, a mobile application, which was further examined in a co-design study with three TVIs and their students. The details of Talkit++ and the co-design study are presented in Section 5.3. We also presented these studies at CHI '19 (Shi et al., 2019).

In Fall 2018, we started rebuilding the design tool. We found that some designers were visually impaired and had trouble using GUI-based design applications. Thus, we designed Molder, a tangible design tool for A3Ms, as describe in Section 6.2. Figure 3.2 visualize the timeline of the projects and their relations to the different sections in this dissertation.

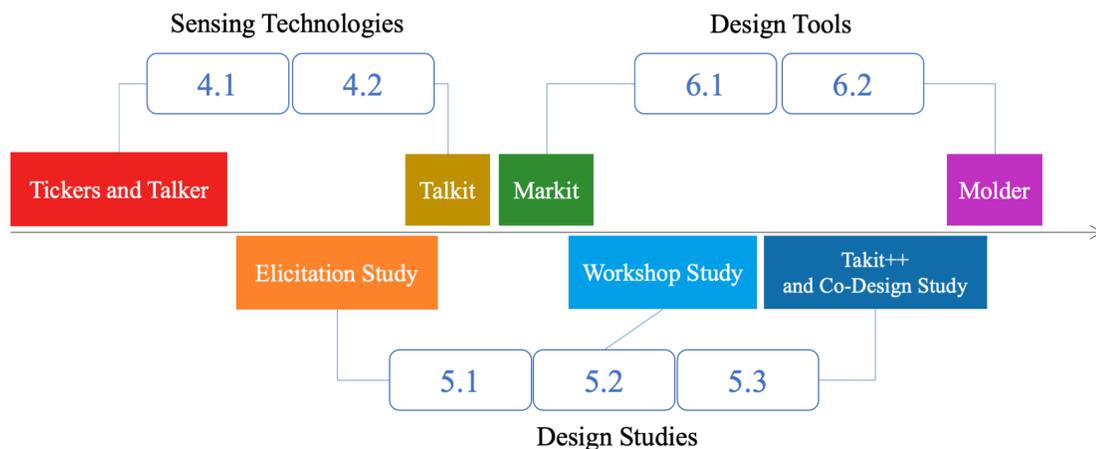


Figure 3.2: The timeline of the projects presented in this dissertation. The projects are labelled with the names of corresponding chapters and sections.

CHAPTER 4

SENSING TECHNOLOGIES

To add interactivity to 3D printed models, this dissertation explores sensing technologies that powered by mainstream devices (e.g., a smartphone) rather than special sensors that need to be set up in a model. As described in Section 2.3, embedded sensors and conductive materials can also augment printed materials with interactivity. However, these techniques would require either fabrication skills (e.g., soldering and wiring) or a lot of modifications to the printed materials. Non-expert designers (e.g., TVIs and O&M specialists) may not have the fabrication and modification skills. To fulfill the design goals in Section 3.1, the sensing technologies should require minimal modifications to printed materials and can be deployed in mainstream devices that are accessible to people with VIs.

Microphones and cameras are two types of common sensors in mainstream devices. We can easily access these sensors from smartphones, laptops, and personal computers. This chapter will start with an acoustic sensing approach using a microphone. Although the acoustic sensing approach achieved a high level of user satisfaction in a lab study, it may be sensitive by the noise in classrooms. Thus, we further explore a computer vision approach using an RGB camera. The computer vision approach is more robust and was chosen as the major sensing technology for the projects in Chapter 5 and Chapter 6.

4.1 Tickers and Talker: An Acoustic Sensing Approach

4.1.1 Introduction

Three-dimensional models are important learning resources for people with VIs. With advances in 3D printing, 3D models are becoming more available. However, unlike visual or tactile graphics, there is no standard accessible way to label components in these models. This work aims to augment models with interactive labels using acoustic sensing.

To add interactive labels on 3D models, we designed a labeling toolkit that includes *Tickers* and *Talker*. We took advantage of acoustic principles and the internal structure of 3D printed models, and created Tickers, small percussion instruments that produce unique tones. Coupled with Tickers, Talker allows people with VIs to access label information by directly touching 3D printed models. To access label information, a user places her phone on the desk and strums a Ticker on the model to produce unique sound (see Figure 4.1). Talker captures the acoustic signal with a microphone and classifies the signal to determine which Ticker was strummed. The application then speaks the corresponding label information. To add label information to a model, a model designer adds different Tickers to a 3D model and uses Talker to record label information for each Ticker while strumming it.

We evaluated our prototype toolkit with nine participants with VIs. In a study with three models, our system achieved a mean accuracy of 93 percent in classifying labels across all participants. We conclude with potential applications and future work.

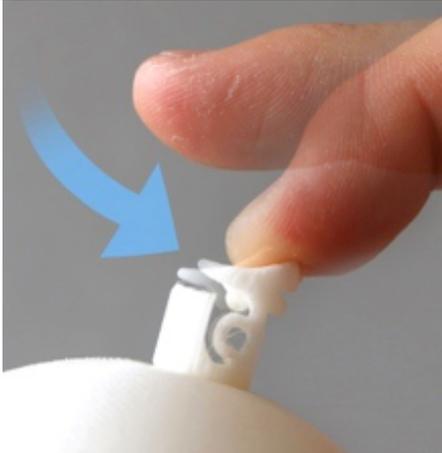


Figure 4.1: A user strums a Ticker.

In this section, we will present the design exploration process along with formative studies. Then, we will describe the design details of Tickers and Talker. Finally, we will show the deployment and results of an evaluation study.

4.1.2 Design Exploration and Formative Study

To design a labeling toolkit, we considered our vision and design goals. As described in Section 3.1, A3Ms should be easy to access, easy to use, and easy to design. To make A3Ms easy to access, we try to avoid complex assembly or expensive auxiliary components. To make A3Ms easy to design, we want to keep the modification process simple. In this section, we will explain different sensing technologies that fulfill these two design goals. In addition to technical exploration, to make A3Ms easy to use, we conducted a formative study with a prototype to solicit user feedback.

We explored different sensors that are commonly integrated into mobile phones. Nowadays, most mobile phones have inertial measurement units, and microphones. We wanted to find a sensor that could sense users' touches while also meeting the design goals (see Section 3.1) and ended up using the micro-

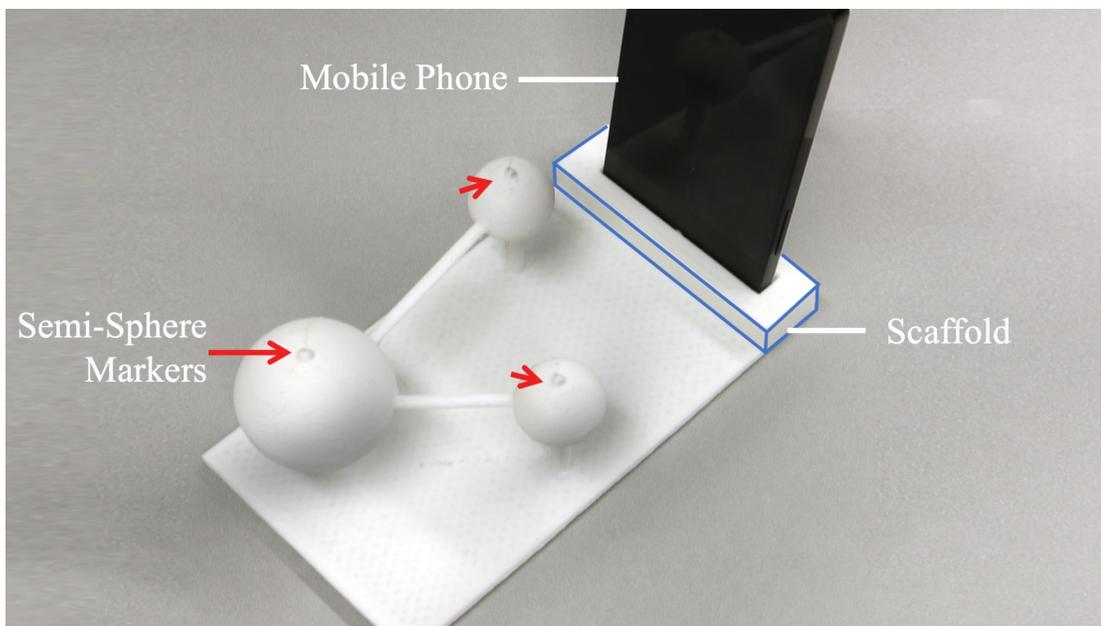


Figure 4.2: The first prototype that uses acoustic sensing. The prototype is designed as a molecule model and has three markers showing label locations and a scaffold supporting a mobile phone that speaks the labels when the markers are tapped.

phone. We considered using inertial measurement units but concluded that it would be hard to classify different labels with noisy data from inertial measurement units. Microphones are cheap and essentially available everywhere. Previous studies showed that when people touch different parts of the same object, the sound they produce from the touch will differ, and that microphones would be sensitive enough to distinguish between the different touch locations (Wang et al., 2014).

We designed an initial prototype (Figure 4.2), which is designed as a molecule model. We chose this molecule model because it is a typical teaching aid. The model has three atoms connected with two bonds. We added semi-sphere markers on the atoms and a scaffold to hold a mobile phone in a fixed

position. The mobile phone detected which marker the user was tapping based on the acoustic signals captured by the microphone. The fixed position of the phone in the model helped the system classify activated labels, since all of the markers were identical. The main difference between the markers was their locations on the model.

We conducted semi-structured interviews to learn about the use of alternative visualization tools and collect feedback about the prototype. We recruited five legally blind participants (4 female, 1 male) for a formative study. The participants' ages ranged from 22 to 55 with a mean age of 33.2 (SD = 12.8). All of them identified as blind. Four participants knew braille, but one didn't use it on a regular basis. Three participants knew Nemeth braille but didn't use it often. All of the participants were smartphone users.

All participants had used some form of alternative visualization tools during their education. They primarily used models in classrooms, while some participants used them when studying on their own at home. However, all of the participants expressed dismay at the availability and access to alternative tools. P3 said: "I didn't use [alternative tools], because they were not available back at that time." They all held high expectations about 3D printing and believed that it would make visualizations more accessible. P1 stated: "[3D printing] is amazing. It can benefit a wide range of blind people, from children to adults."

Surprisingly, several participants who knew braille preferred to have a sighted person read labels in visualizations, rather than reading braille labels. They explained that braille changed the texture of the surface, making it hard to distinguish between the model and the labels. Second, braille required a large surface area, limiting the length of each label and the overall number of

labels that fit onto a model. Although abbreviations are used to compensate for braille’s bulkiness, P5 expressed that she likes “having things labeled as much as possible” and that “[an abbreviated label] doesn’t seem like enough information.”

We proceeded by introducing our first prototype to our participants and explained the idea of using a smartphone to increase the accessibility of labels on 3D models. Since participants joined our study at different development stages, we presented them with our up-to-date prototype at the end of the study and collected feedback to evaluate our system and inform future design iterations.

For our earlier prototypes, a few participants seemed confused about how to tap the markers and pushed down on the marker instead. Although this problem could have been addressed by training the users, we learned that the markers did not have clear affordances for tapping. A few participants also found that some of the markers were difficult to locate, as they weren’t sure whether the marker was part of the model or not. These findings validated the need to create markers that were easy to distinguish from the rest of the model. Overall, despite some confusion, all participants expressed their interests in using our interactive models.

We identified two main design challenges after observing how some participants interacted with our first prototype:

First, tapping sounds were not consistent across users. Participants used different parts of their fingers to tap the markers, which caused different acoustic features. Although there were some studies about acoustic sensing, most of them used personalized machine learning models, where the end-user trained

the acoustic sensing system (Harrison et al., 2010; Sato et al., 2012; Laput et al., 2015). While recent work showed that we could train an acoustic classifier using online videos and sound effect libraries (Laput et al., 2018), there is no suitable sound dataset for our use case. In our work, the person who trains the models (a model designer) is not necessarily the same person who uses the model. It's important to keep sound consistent so different users can access labels even though the model is trained by one person.

Second, participants wanted to explore the models more freely. Before accessing the label information, participants with VIs spent time exploring models by holding them with both hands. A scaffold with a fixed mobile phone made this exploration process difficult.

4.1.3 The Design of Tickers

Considering the design challenges in our system, we explored different auxiliary component designs. We aimed to find one solution that would be more accurate in classifying labels while allowing users to explore the model without having to affix the phone. To do this, we designed markers with unique acoustics that were consistent across all users (i.e., independent of the user's touch behavior). We applied basic acoustic theories to our designs.

4.1.3.1 Acoustic Signals and Percussion

The acoustic signals of musical instruments are complicated and can be influenced by many factors. Percussion instruments have unique musical tones that can be distinguished from each other. A musical tone is determined by its du-

ration, loudness, pitch and timbre (Roederer, 2008). A musician can change the duration and loudness of a musical tone when he plays the instrument at different strengths, for instance by banging a drum with different amounts of force. On the other hand, for an instrument like a drum, the pitch and timbre are not so dependent on the musician playing the instrument. Both pitch and timbre are psychoacoustic terms and usually quantified with Fourier Transform (Bracewell, 1986) in signal analysis. Two properties affect the tone of the sound: (1) the internal structure of instruments will change the pitch and timbre; (2) sound changes along the way it travels, and it depends on the distance and the medium through which it propagates.

1. Structure Property: Internal Structure

We can alter the pitch of an instrument by changing its enclosed air space (Rossing, 2000). Most percussion instruments rely on the vibration of air to create sound. When we change the enclosed air space, we also change the patterns of air vibrations. One practical example is that when we pour water to a glass, the pitch of the sound will increase gradually.

2. Path Property: Sound Propagation Path

Sound is a type of mechanical wave that propagates in air, fluid, or solid media. Like all waves, it has constructive and destructive interference over its path as it travels through space. Even with the same sound source, different traveling paths will make the perceived sound different at different locations (Berg, 1995). One extreme example of this is that when we dive into water, the sounds we hear are different from what we would hear above the water.

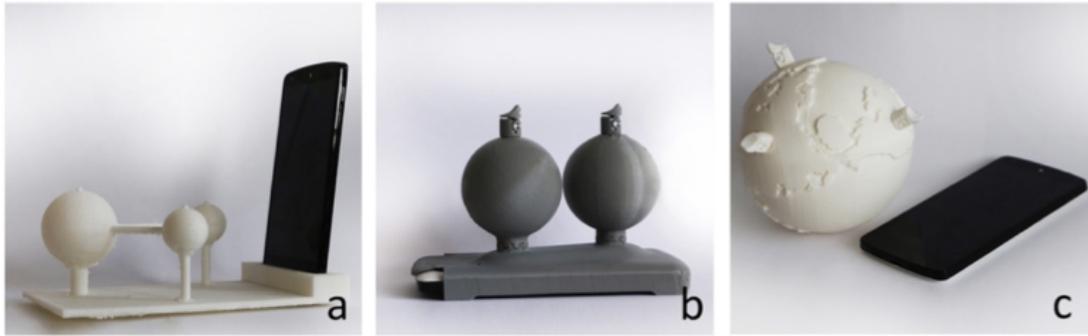


Figure 4.3: Three prototypes that uses acoustic sensing. (a) The first prototype required a fixed mobile phone propped in the scaffold. (b) The second prototype included a case for a mobile phone. (c) The third prototype didn't require a fixed mobile phone; it could be placed on a desk as the user held the model.

These two properties are related because sound travels through an instrument's body, which affects its acoustic properties

4.1.3.2 Model Iteration

Before redesigning our markers, we needed to decide where to put our microphone. On the one hand, we could get steady signals with a fixed microphone according to the Path Property. On the other hand, we wanted to enable users with VIs to explore the model freely without trouble. Thus, in our second prototype, we put the mobile phone on the base of the model and made the scaffold into a phone case. However, embedding a mobile phone into a model made the model heavy and constrained exploration. In our third prototype, there is no base required and users could explore models easily. Model designers only need to add different markers, and users can experience the model without an added base. These prototypes are demonstrated in Figure 4.3.



Figure 4.4: Ten markers with a quarter for scale. On the top left is a Ticker.

4.1.3.3 Tickers

Instead of adding a base, we improved our markers to get a more consistent sound. We explored different markers to find a design that can produce unique sounds across types of markers, but consistent sounds across users for each type of marker. Figure 4.4 shows ten designs of markers. These designs were inspired by different percussion instruments and handicraft tools. Percussion instruments can generate unique and consistent sounds while handicraft tools have ergonomic designs that support strong affordance. We brainstormed marker designs with different lengths, cap designs and sizes. In order to test different markers, we designed them as screws so we can exchange them without reprinting a model. They can also be printed in a model, so they don't require any assembly.

We tested these markers and chose the one named Ticker, which has a hol-

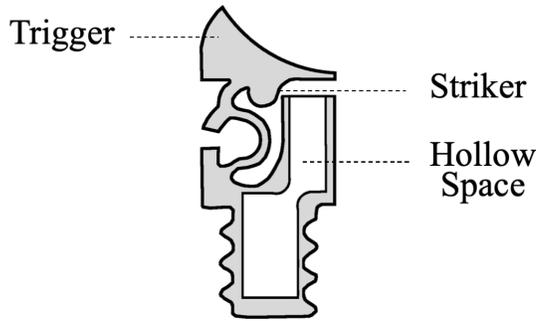


Figure 4.5: The structure of a Ticker.

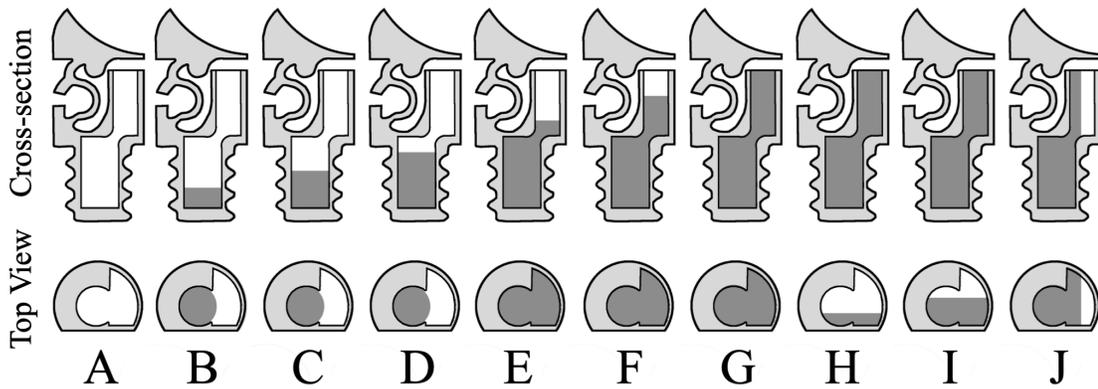


Figure 4.6: Ten markers with a quarter for scale. On the top left is a Ticker.

low space and strong affordance (see Figure 4.5). A Ticker has a trigger, which is a curved surface. We added a stopper to prevent users from breaking the Ticker when strumming it. To strum a Ticker, a user pushes and then releases the trigger. When the trigger goes back, the striker hits the hollow space and generates a unique sound. Changing the size and shape of the hollow space changes the pitch of the produced sound. in

We designed ten Tickers with different internal hollow structures, as shown in Figure 4.6. From Ticker A to Ticker G, we incrementally added 2mm of padding to the hollow space. From Ticker H to Ticker J, we cut the padding vertically in different ways. Strumming these Tickers produced sounds that were distinct to our ears.

While these Tickers are produced through an empirical design process, current and future work in digital fabrication and acoustic sensing can help us refine these designs. Tickers were designed through brainstorm and testing. This process relies heavily on researchers' experience and skills. Recent work showed that this process could be automated. For example, Tejada et al. (2018) demonstrated that it was also possible to computationally design the acoustic feature of a printed object. In addition to the automated design process, future work also needs to verify the influence of printing settings. Theoretically, a Ticker should have a consistent sound as long as it is made of a certain material, regardless of its printing settings (e.g., types of printers, printing orientation). However, we did not confirm this through experiments. We optimistically think that future fabrication technologies will be able to create identical objects that have a same design.

4.1.3.4 Formative Evaluation of Tickers

We conducted a study to explore whether our Tickers were capable of creating different acoustic features using the Structure and Path Properties.

To examine the two properties, we measured the acoustic similarity between different Tickers in the same location and the same Tickers in different locations. We created 12 Ticker-location combinations using the molecule model shown in Figure 4.7. We first inserted Tickers A-J into the Oxygen atom (denoted as A-O to J-O), and then inserted Ticker J into each Hydrogen atom (denoted as J-H₁ and J-H₂). We tested the Structure Property by computing the acoustic similarity between all pairs of Tickers that were inserted in the same location (A-O to J-O) and tested the Path Property by computing the acoustic similarity between



Figure 4.7: An H_2O molecule with three atoms labelled with Tickers.

$J-H_1$ and $J-H_2$. In music, similarities between two samples is often measured using the Euclidean distance between their amplitude spectral densities (ASD) (Tzanetakis and Cook, 2002). A smaller Euclidean distance means higher similarity. The ASD is the energy distribution of signal over different frequencies and is calculated using a Fast Fourier Transform (FFT).

We recruited five sighted participants (3 male, 2 female) from our campus. The participants' ages ranged from 21 to 28 with an average of 23.4 ($SD = 2.7$). During the experiment, we asked each participant to strum the Ticker 30 times for each of the 12 Ticker-location combinations. We recorded samples using the bottom microphone of a Nexus 5 smartphone at a 44.1 kHz sampling rate.

We ran an offline similarity analysis with the recorded samples. Each participant produced 360 samples. We computed the ASD for each sample, and, for each participant, computed the average Euclidean distance between all pairs of samples. We plotted the results in five confusion matrices (i.e., one confusion matrix for each participant). Figure 4.8 shows one example. Observing pairwise comparison of A-O to J-O, we found that different Tickers had different tones,

which accorded with the Structure Property. Also, looking at pairwise comparisons for J-O, J-H₁ and J-H₂, we found that the same Ticker could generate different sounds when inserted in different locations, which confirmed the Path Property. Thus, although we only provided ten designs of Tickers, designers can combine them with different locations to create unlimited unique sounds. We do acknowledge that Path Property might be weaker than Structure Property in certain situations. For example, different locations may lead to a same path but different tones can be distinguished with a good classifier. In sum, our Ticker designs were effective because different Tickers had acoustically distinct tones, especially when placing them in different locations on a model.

4.1.4 The Design of Talker

We designed Talker, a signal processing application that can make and access interactive labels with Tickers. A model designer can use Talker to record Ticker acoustic samples and associate an audio label with each Ticker. Users with VIs can access a label by strumming a Ticker three times.

Talker's algorithms consist of two parts: (1) detecting the signals of a strum from background noise and (2) classifying the detected signals as one of the model's Tickers.

4.1.4.1 Strum Detection

Talker continuously captures and processes sound from a microphone. When creating and accessing labels, users strum the Tickers whenever they are ready, and Talker will detect the strums from other sounds.

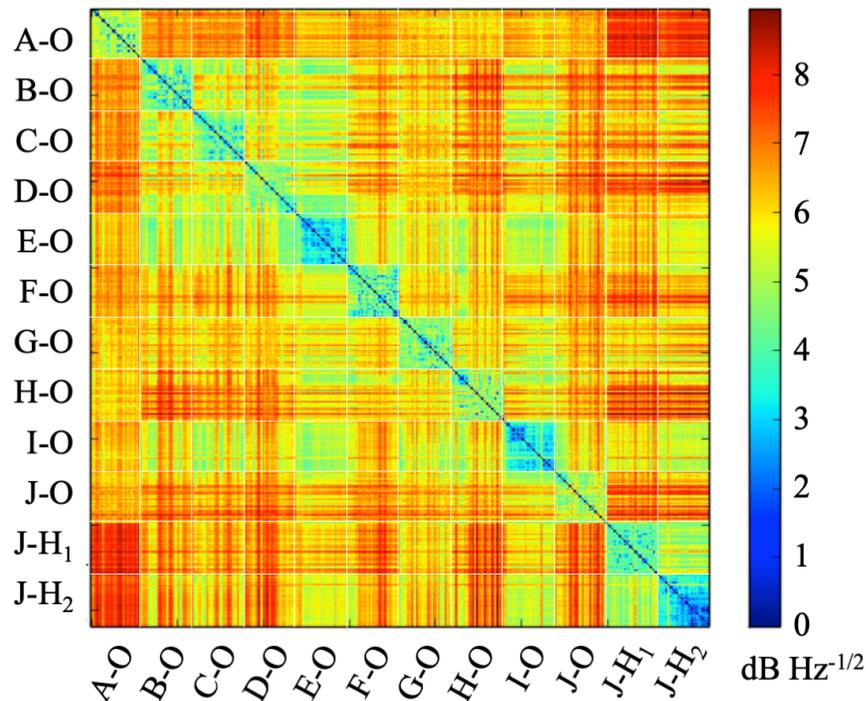


Figure 4.8: A matrix showing the similarity between Ticker tones in a molecule model. The similarity is measured by the Euclidean distance between Ticker tones' amplitude spectral densities. Lower values means more similar. Tones from different Ticker types and locations are relatively distinct, which is promising. For examples, pairwise comparison of A-O to J-O (i.e., different Ticker types) has a high value. Pairwise comparisons for J-O, J-H₁ and J-H₂ (i.e., different locations) have high values. In an ideal situation, the center diagonal would be dark blue, and the rest of the matrix would be dark red.

The first step in deciphering the user's input is to detect which part of the continuous signal corresponds to a strum. Our system uses a *median filter* and a *root mean square (RMS) filter* to filter out human voice and background noise. The median filter is widely used in signal processing and proven to be effective in detecting impulsive sound signals. A median filter continuously computes the median of a moving window, and outputs results based on the median. For a median filter with W -point moving windows, the filter will significantly

decrease the strength of pulses whose duration is lower than $\frac{W}{2}$. Because a strum only lasts 0.1 seconds, it is quite easy to filter out human voices with the median filter when we set W to a relatively small value. For a detailed analysis of the median filter, see prior work by Dufaux (2001). We optimized the median filter algorithm so it can run continuously on our computer and mobile devices.

We use the median filter to indicate the start of a strum, so we send the following 0.1 second of data to the RMS filter. RMS filters delete short peaked noise (e.g., an accidental scratch on a table). The RMS filter calculates the RMS of the captured 0.1s signals. The RMS represents the energy of the signal. The RMS filter deletes signals that have an RMS that is below our threshold.

4.1.4.2 Signal Classification

We used the ASD components of a sample, which we calculated with an FFT, as classification features. As noted earlier, the ASD is the energy distribution of signal over different frequencies. Our preliminary results showed that high-frequency ASD components had a negative effect on classification accuracy. Thus, we empirically chose a frequency range of 0 Hz to 10000 Hz. At a sampling rate of 44.1kHz, this results in 1000 features for each 0.1s sample.

We used the three-nearest neighbor classifier, which classifies signals based on the similarity between the signals and samples. For each input signal, the algorithm computes the Euclidean distance between the signal and each recorded sample and finds the three closest samples to the signal. Based on these three samples, the algorithms output which Ticker was most likely strummed.

To make the detection process more robust, we require users to strum a

Ticker three times to activate it. The system classifies each strum individually and outputs the final result using a majority rule. If at least two of the three strums are classified as the same Ticker, Talker outputs the label for that Ticker. If no Ticker satisfies the above condition, the system does not output anything, and the user must try again.

To record information, a user strums a Ticker 50 times and then records the voice label with Talker. In our prototype, the application associates and stores the data in arrays.

4.1.5 Evaluation

Our system has two target user groups: designers who create A3Ms with appropriate Tickers and users with VIs who access labels. As the first step towards A3Ms, we focused on a user's ability to access labels. We conducted an evaluation study, where we aimed to understand (1) how often our system spoke the expected label and (2) the overall user experience of perceiving a physical 3D model with its label information.

4.1.5.1 Method

Apparatus

We built a prototype of Talker on a Nexus 5 and a Macbook laptop. The mobile phone captured audio at a 44.1 kHz sampling rate, and streamed audio signals to the laptop. The laptop acted as a server and processed the data. We programmed the server in Python and used classification algorithms from the

Scikit-learn library (Pedregosa et al., 2011).



(a) a globe with labeled continents



(b) a molecule with three labeled atoms



(c) a terrain map with labeled locations



(d) three labeled animals

Figure 4.9: Four printed models labeled with Tickers.

We designed and printed four models, as shown in Figure 4.9. We designed our models in SolidWorks (2019) and printed them on a Makerbot 5th generation printer set to default settings (15% infill, 0.2 mm layer height) with filaments from Makerbot (2019). The four models are:

1. *Globe*: A globe model (Figure 4.9a), we added audio labels for “Asia,” “Africa,” and “America.” We only used this model for teaching and train-

ing purposes.

2. *Molecule*: A H_1O model (Figure 4.9b). Since there are two Hydrogen atoms in the model, we put a small sticker on the surface of one atom so that the researcher can differentiate it visually from the other one. We added voice labels for “Oxygen,” “Hydrogen,” and “Hydrogen with a sticker.”
3. *Terrain*: A terrain map (Figure 4.9c). We recorded labels for “Mountain,” “River,” and “Plain.”
4. *Animals*: Three animal models (Figure 4.9d) with three labels: “Dog,” “Deer,” and “Elephant.”

We chose these models because they are common in STEM education and they represent different spatial structures. The Globe model is one single object with several labels, while the Animals model are a set of objects. The Terrain model is a large 2.5D model, while the Molecule model has a symmetric 3D structure. These models can help us understand how Tickers and Talker, along with the Path Property and the Structure Property, work in models that students with VIs use in their classrooms.

For each model, we used Ticker E, Ticker G, and Ticker I as in our offline study. For each Ticker, a researcher recorded 50 training samples and an audio label. Before handing the model to a participant, the researcher indicated which model was being used in the Talker prototype application.

To facilitate testing and the studies, we printed the Tickers and models separately and inserted the Tickers after printing. In practice, users could print the models along with the Tickers. In our fabrication process, we did encounter some challenges when printing the models along with the Tickers, as we wanted

to avoid printing supports inside the cavities of Tickers. There are two potential solutions: one is to carefully arrange the orientation of Tickers to avoid supports. The other one is to use water-soluble materials (e.g., PVA (Ultimaker, 2019)) to print easy-to-clean supports.

Participants

We conducted a user study with nine participants (5 male, 4 female), with ages ranging from 22 to 81 years, and an average age of 38.4 (SD = 18.5). Eight participants had at least college degrees, and one participant had high school degree. All participants were legally blind. One identified as low vision, while the remaining identified as being blind. Most participants were familiar with braille. Seven participants knew braille but three of them didn't use braille regularly. Three participants had used Nemeth braille before. However, most participants did not have much experience with tactile graphics or tactile models and none of them had used them recently. Eight participants owned iPhones. Five of them participated in our formative studies. We compensated participants with 15 USD per hour and reimbursed them for transportation expenses up to 60 USD.

Procedure

The study consisted of one session that was approximately 60 minutes long. After introducing the project to a participant, we (1) taught her how to activate labels by strumming a Ticker, (2) asked her to repeatedly activate labels for the three models and (3) conducted a short user experience interview to gather subjective feedback about our labeling method.

During the first 15 minutes of each session, we taught participants how to

activate a label on a model. We began by describing the shape of the Tickers and explained how to strum them. We then put a mobile phone with Talker installed on a table next to the participant and handed her the Globe model. We told her to strum each Ticker three times to activate the corresponding voice label. After some practice, participants were asked to activate each Ticker once, and then repeat the process a minimum of eight times, until they felt comfortable enough to activate Tickers without our assistance.

After practicing activating labels on the Globe model, we asked each participant to access labels on the Molecule, Terrain, and Animals model for our formal evaluation. We presented our participants with one model at a time, asked them to explore the model to get familiar with it, and then activate each Ticker on that model once. We then repeated this process for a total of eight rounds per model, such that each Ticker was activated eight times. Participants took a break after exploring each model. We counterbalanced the order of the models across users with a Latin Triangle.

We conducted a short interview to collect subjective feedback from our participants. We asked them to rate nine statements (S1 S9) on a scale from 1-7, where 1 was strongly disagree and 7 was strongly agree:

S1: I would like to use this system.

S2: The system is easy to use.

S3: I would need the support of a sighted person to be able to access labels.

S4: I would imagine that most people would learn to use this system very quickly.

S5: The system is very cumbersome to use.

S6: I needed to learn a lot of things before I could get along with this system.

S7: It's hard to identify the markers (Tickers) on the models.

S8: It's easy to strum the markers (Tickers).

S9: The models were hard to understand because of the markers.

We mixed positive and negative statements intentionally to reduce acquiescence and extreme response biases. We also asked participants to explain the reasons behind their scores.

Design and Analysis

The study was a within-subjects design. For each participant, they activated 72 ($3 \times 3 \times 8$) Tickers in total. We recorded which Ticker a participant tried to activate, and the result of the audio label. To assess the performance of our system, we analyzed its classification results using descriptive statistics. We also transcribed participants' comments and identified common themes.

4.1.5.2 Results

Accuracy

Our system achieved a mean accuracy of 93 percent ($SD = 0.04$) in classifying activated labels across all participants. All participants achieved a mean accuracy of at least 88 percent for three models, as shown in Figure 4.10. One participant, P3, achieved the highest accuracy at 100 percent. Meanwhile, P4 and P9 had the lowest accuracy with a mean of 88 percent. Both of them had long fingernails, which may have affected the acoustics of their strums, although further research

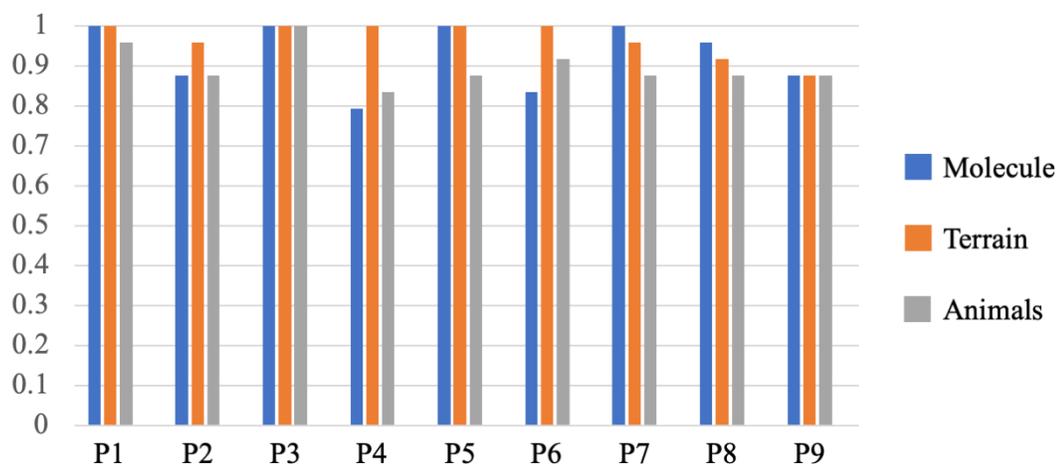


Figure 4.10: The mean accuracy for each participant and each model. Higher values are better.

would be needed to confirm this. Our oldest participant, P2, who was 82 years old and completely blind, managed to complete our study and achieved a mean accuracy of 90 percent, which indicates the system is easy to learn.

When evaluating the performance of each of the models, we found that Terrain was the most stable and accurate across all participants, while Molecule and Animals models didn't work well for some participants. In Figure 4.11, we see that Tickers on the Molecule and Animals models were confusing. This was especially the case for the "Hydrogen 1" and "Hydrogen 2" Tickers on the Molecule model and all of the Tickers on the Animals model.

Subjective Feedback

We present the mean scores and standard deviation of scores for each of the nine statements (S1 - S9) in Figure 4.12. In general, all participants were positive about our system. However, three participants (P2, P3, and P6) felt the system was not easy to use (relatively negative scores for S1 and S5). They were not

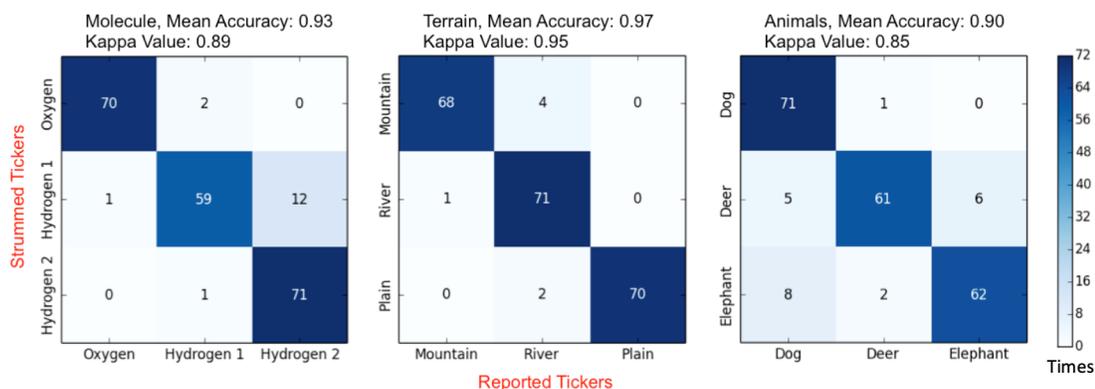


Figure 4.11: The mean accuracy of the three models and misclassification rates for each of their Tickers. Nine participants activated each Ticker, producing a total of 72 data points. Tickers on the Molecule and Animals models were confused more frequently than Tickers on the Terrain model.

entirely content with Tickers and expressed a desire to improve them by making them smaller and easier to strum (less stiff). P2, the 82-year-old participant, explained, “because it’s important to know how to aim it, how to strum [it].” P6 also said it was difficult to strum the Tickers (low score for S8) and attributed this to the “discomfort of the markers [Tickers].”

When asked whether and how they would like to use our interactive models, several participants noted that our interactive models would be suitable for educational purposes. The participants also expressed that they wish they had access to a system like ours to help them learn new concepts. P4 said, “If I went back to school, I would be very excited [about your interactive models].”

Most participants felt confident about being able to distinguish Tickers from the rest of the model. However, P3, P5, and P7 were concerned about mistaking Tickers with other elements, such as mountains on Terrain because they felt that the current size of the Tickers significantly changed the overall topography of

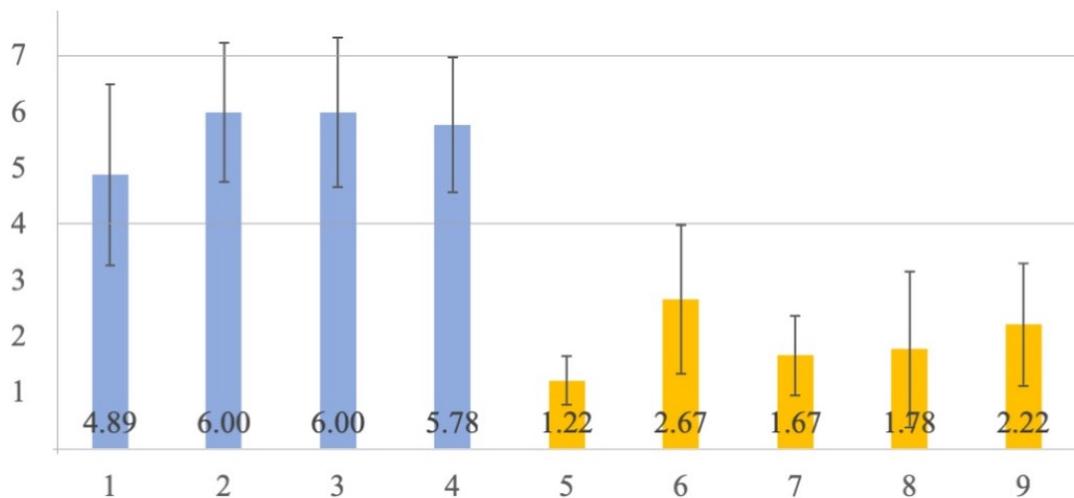


Figure 4.12: Mean scores for subjective statements. 1 stands for strongly disagree and 7 stands for strongly agree. Error bars represent 1 standard deviation. For positive statements (blue; S1, S2, S4, S8), higher values are better. For negative statements (yellow; S3, S5, S6, S7, S9), lower values are better.

the model. P3 explained that her low score for S7 (hard to identify Tickers) was “only because of the terrain map model, everything else [the other models] was fine.” Reflecting on how some models would be more effective in providing tactile visualization information to users, P5 pointed out that the helpfulness of using our system “could go either way, it will depend on the model itself.” He further explained that Tickers were distinct in the Animals model but could be confused with elements in the Terrain model.

4.1.6 Discussion

Overall, our system achieved promising results. With only 50 training samples for each Ticker, we achieved label classification accuracies that ranged from 79

percent to 100 percent for each of our nine participants and three models. The Terrain model achieved an accuracy of 97 percent across all participants. Users were able to explore the models effectively and they were excited about having labeled 3D printed models.

We observed that participants learned how to use our system quickly and were able to locate and activate Tickers with ease. When we handed participants a new model, they strummed each of the Tickers within about 20 seconds. It only took about two seconds to activate a single Ticker and access an audio label.

We think labels were sometimes misclassified because of the physical configuration of the models. In the Molecule model, the Tickers for labels “Hydrogen 1” and “Hydrogen 2” were confused by the system because of the symmetrical structure of the H₂O molecule. According to the Path Property, because the two atoms are in a symmetrical structure, their sounds tend to be same. Although we used different Tickers, the difference between these Tickers is not distinguishable enough to the trained classifier. A better classifier may achieve a higher accuracy in this situation. In the Animals model, each animal had a similar body shape. Thus, the Path Property was not applicable in these cases. The other models had irregular shapes, so, following the Path Property, their Tickers produced more distinct tones. In general, the sound difference would be more distinguishable with different Tickers on different locations.

According to our results, the effect of the Structure Property on acoustic features was not as strong as the Path Property, especially in the cases of Molecule and Animals models. Both the Animals model and the Molecule model were quite small, so users easily influenced the Tickers tones on them. Even though

each Ticker generated a somewhat distinct sound, the way users held the models and the way they strummed the Tickers modified the sound, varying it across different trials. The acoustic features of our Tickers were not distinct enough to fully take advantage of the Structure Property.

Although all participants were positive about using our system, several suggested the limitations of Tickers. Several participants noted that they felt tired after using Tickers for a long time. We also noticed that participants confused Tickers with other elements. To address this issue, we could embed Tickers the model (i.e., expose only the Ticker trigger on the surface of a model). However, this could potentially make it harder for users to locate Tickers to activate labels. We will keep these trade-offs in mind for future iterations of Tickers. Another solution is adaptive Tickers. We could develop Tickers with different shapes of triggers, and designers can choose one type of Tickers to avoid confusion.

The evaluation study has several limitations. We did not evaluate Tickers and Talker in an education setting with target users (e.g., students with VIs). Instead, we only tested the classification accuracy and general usability of the system with adults with VIs. The participants were instructed to access labels on the models we prepared and did not fully explore the models by themselves. We think our study is an important first step to show the potential of A3Ms. However, future study should further examine A3Ms in education settings with target users and allow users to naturally explore A3Ms.

Tickers and Talker, as an acoustic sensing approach, demonstrates the potential of A3Ms and motivates us to explore a more robust sensing technology to augment printed materials. Although Tickers and Talker achieved promising results, we found there were some limitations about this approach. First, the

evaluation results indicated that the accuracy and bandwidth of acoustic sensing technology were limited and adding more labels would inevitably decrease the accuracy of the acoustic recognition system. Second, it would be difficult to add many Tickers on a small model considering the size of Tickers. Third, Talker is sensitive to noise in classrooms. Thus, we need to use a sensing technology that can support multiple interactive elements and is robust to the dynamics in classrooms.

4.2 Talkit: A Computer Vision Approach

4.2.1 Introduction

To address the new challenges we discovered in Section 4.1, we designed *Talkit*, a computer vision application that enables users with VIs to interact with augmented models. The application only needs a commodity camera and a microphone, without any needs for complicated assembly, additional electronics, or special materials. To create an A3M, a model designer (e.g., a TVIs, an O&M specialist) marks model elements and associate them with text annotations using a design tool, which will be explained in Chapter 6. To begin exploring a printed model, a user with VIs sticks a sticker on her fingernail and slides a printed tracker on a printed scaffold (Figure 4.13). Using an RGB camera, Talkit locates the model by sensing the tracker, and finds the user's finger by tracking the sticker. Talkit speaks audio information when the user puts the finger with the sticker on an annotated element on the model. The user can also access more information using a speech command. The tracker and stickers are printable and reusable.

We evaluated Talkit in a study with participants with VIs. Our results showed that participants with VIs found all annotated elements on models with five annotations in an average of 64 seconds ($SD = 40$), and they found a specific annotated element in an average of 7 seconds ($SD = 8$).

In this section, we will start with a brief description of an exploratory study. Then, we will describe the design and evaluation of Talkit. In Section 5.1, we extended the exploratory study and did more thorough analysis to include find-

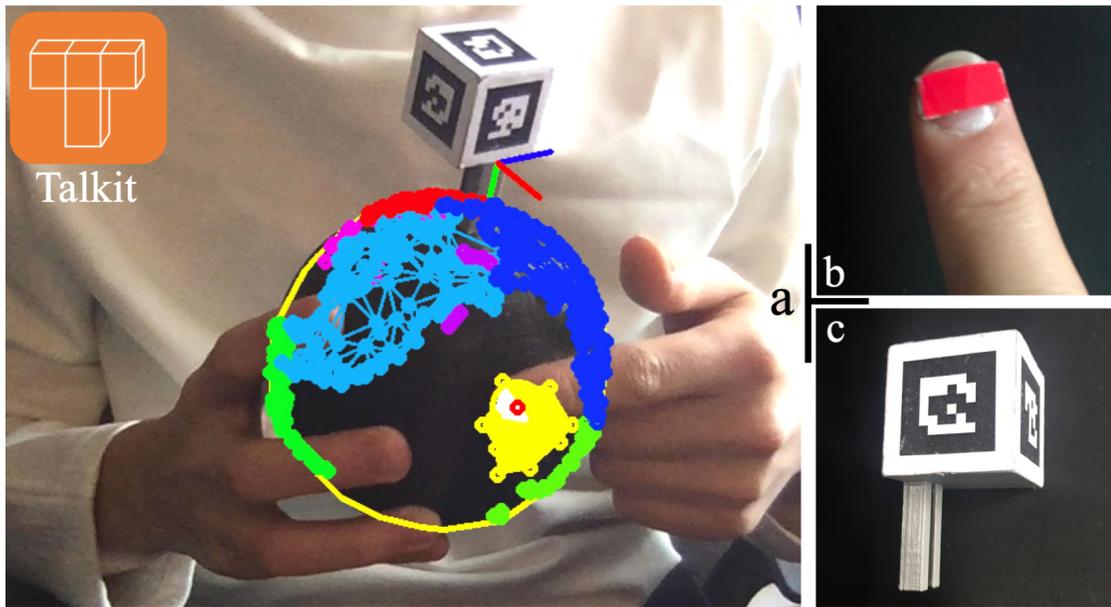


Figure 4.13: A user with VIs interacts with a globe model using Talkit. She sticks a simple sticker (b) on her fingernail and slides a tracker (c) on the printed scaffold. She then launches the Talkit application (a) on her computer and interactively explores the model with its audio annotations.

ings about exploration behaviors and user-defined input techniques. In Section 6.1, we will report a design tool that helps people to create A3Ms for Talkit.

4.2.2 Exploratory Study

We conducted an exploratory study to develop effective interaction techniques for accessing annotations from tactile models. We aimed to understand what interactions people with VIs prefer to use when exploring A3Ms. For the full details and analysis of the study, see Section 5.1, which includes more findings that may benefit other researchers. In this section, we will highlight the findings that led to the design of Talkit: (1) finger gestures and (2) flow control.

We recruited 12 participants with VIs (8 females, 4 males) whose ages ranged from 23 to 60 with a mean of 40.75 (SD = 13.15). All were legally blind. One identified as low vision, while the remaining identified as blind. Eleven participants knew braille, but four of them didn't read it on a regular basis. Eleven participants used iPhones, and the remaining participant did not use a smartphone. During the study, we introduced the idea of A3Ms and asked each participant to design different gestures and interaction techniques. We asked them to explore three models we prepared and suggest methods for accessing audio annotations.

All participants except for P12 shared the idea of using finger gestures from touchscreens to interact with models. The most common gesture was tapping on an element (e.g., a continent, a building) to retrieve its annotation. Even P5, the one participant who did not have smartphone experience, wanted to touch tactile elements of a model to access annotations. P7 explained that “[using gestures from smartphone interaction] makes accessibility consistent across platforms.”

Participants emphasized the importance of being able to control the flow of the audio output. When exploring an element, they wanted to know holistic information first and then hear details if desired. Three participants also talked about adding buttons to switch the system on or off. P5 said the system should be “off at the beginning, and then you can feel for yourself without talking.”

4.2.3 Talkit

We designed Talkit based on the design goals (see Section 3.1) and findings from the exploratory study. To lower the access barrier of A3Ms, we decided to use computer vision technology to avoid the additional materials used in most prior work. Nowadays, cameras are available in all kinds of smart devices. We chose RGB cameras as our start point because they were more available than depth cameras in commodity devices people with VIs were using (e.g., iPhones)¹. Since a computer vision-based solution doesn't involve attached electronics, makers can easily annotate existing models with simple modifications. However, computer vision technology can have disadvantages as well, such as high computation demands and a camera's limited visual field. By using commodity RGB cameras, we may face some trade offs. For example, the complexity of gestures is limited due to the lack of depth information, and the system can be influenced by lighting conditions and ambient noise. We considered these limitations as we iterated on our design.

Talkit allows users to interact with the physical models with a pointing gesture and speech input. Talkit has three components: (1) a speech command system, (2) an audio guiding system, and (3) a computer vision system. We describe these components in this section.

¹As mentioned in Section 3.3, we started building Talkit in Spring 2016. Back then, depth cameras were not as available as today.

4.2.3.1 Interaction Flow

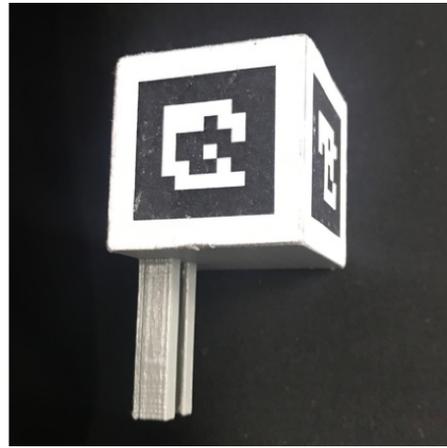
To access the audio annotations on a model with Talkit, a user follows these steps:

1. Before interacting with the model, a user attaches a sticker to her fingernail (Figure 4.14a), slides the tracker (Figure 4.14b) into the printed scaffold, and enters a model ID (using a keyboard).
2. The user positions the model and the finger with the sticker within the camera's field of view by following the instructions from the audio guiding system. When a model is out of the camera's field of view, the audio guiding system reminds the user to reposition the model by providing specific directions, such as "move up." If no finger is detected on the model, the system asks the users to put her finger on the model.
3. The user points the finger to an element of interest and holds until Talkit speaks audio information associated with the element. We empirically set the holding time to 0.2 seconds for the current prototype.
4. She can also control the audio output with speech input, such as saying "more" to get more detailed information about the pointed element. She can also turn on or turn off the audio using speech commands.

In the following section, we introduce Talkit's computer vision algorithms.



(a) a red sticker



(b) a printed 3D tracker

Figure 4.14: Two required accessories for Talkit.

4.2.3.2 Algorithm Design

Locating A Model

There are three coordinate systems involved in Talkit. In a digital model, all faces are stored in its *model coordinate system*. When printed, a digital model becomes a physical model, which interacts with other physical objects (e.g., fingers) in a *world coordinate system*. Talkit uses an RGB camera to capture these interactions in a video stream, which represents a *2D projective coordinate system*.

Talkit locates a model in the current 2D video according to the position of the 3D tracker and its relative position on the model. Talkit defines a 3D point of the tracker scaffold as the origin of a world coordinate system and can also achieve the point's position in the model coordinates based on a digital model's data. Based on these coordinates, Talkit transforms all the faces of the model into the world coordinate system. Talkit then maps the world coordinate system into the projective coordinate system by recognizing the current position of the

tracker. After transforming all faces into the projective coordinate system, Talkit uses a Backface Culling algorithm (Zhang and Hoff, 1997) to eliminate faces that are occluded from the camera to solve the occlusion problem and reduce its computational load.

Identifying Activated Elements

After locating a model in the video stream, Talkit tracks the position of a user's finger to identify the element that she wants to activate. Talkit tracks the finger by recognizing the sticker on the user's fingernail. In prior work, we originally used a skin-color detector to track the finger (Shi et al., 2016a). However, the system could not identify the "pointing finger" when a user explored the model with multiple fingers, so we used the sticker instead.

Within the range of the model in the video stream, Talkit uses an HSV color-based detector to locate the fingertip. Then, it finds the area of the specific sticker color within the model's outline, and then calculates the center point of the largest contour as the location of the fingertip. In this way, the system can quickly locate the finger and isn't confused by the colors in the surrounding environment.

Then, Talkit finds model faces near the location of the fingertip, as shown in Figure 4.15. The system first identifies one key face under the center point. If there are more than one front faces under the center point (e.g., front-facing faces that are occluded from the camera), the system uses a Z-buffering algorithm to find the face closest to the camera and regards it as the key face. Since a fingertip occupies a relatively large area on some models, Talkit also considers the faces that (1) are close to the key face and (2) overlap with a circle around the center

point. This algorithm can quickly find faces that are close to the finger location and avoid activating occluded front-facing faces.

Talkit identifies activated hotspots in the found faces and plays the audio annotation. If there is no hotspot in the found faces, the system says “unmarked” to indicate that the user should move on. If all activated hotspots belong to one element, the system speaks the label of the element and speaks a detailed description when the user says “more.” If there is more than one element in these activated faces, Talkit will speak the number and the name of the elements. The user can also retrieve information about the relative positions of these elements by saying “more.” For example, when a user points to a continent on a globe, the system says the name of the continent (e.g., “North America”), and speaks the detailed description of the continent after the user says “more.” When the user points to the border of “North America” and “South America” on the model, Talkit says “Two, North America and South America.”

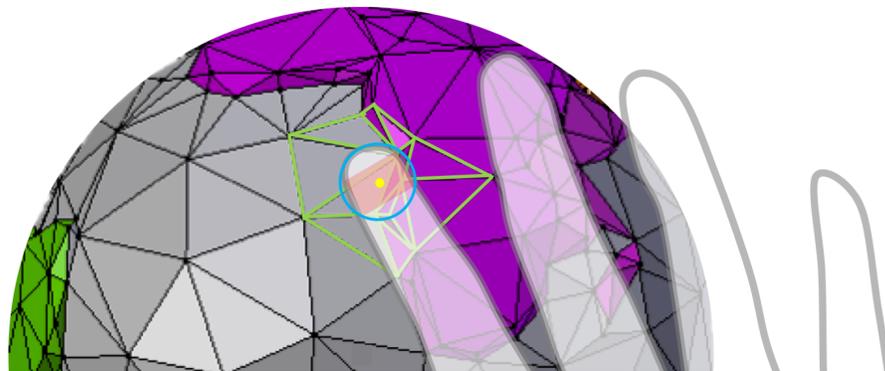


Figure 4.15: A user points her finger on the edge of the purple hotspot. The Talkit application identifies the center point of the sticker (marked in yellow dot) as the position of the finger and finds faces (outlined in green) that overlap with a circle (outlined in blue) around the center point.

If the user asks for further information, the system will say “North America up, South America down.”

4.2.3.3 Implementation

Talkit was implemented in Python. We used OpenCV (Bradski and Kaehler, 2008) for real-time video processing. The Talkit application used the built-in calibration function from OpenCV to correct camera distortion. A camera only needs to be calibrated once for all camera-related applications. To map the 3D information of a model to a 2D coordinate system, we used the Chilitags library (Bonnard et al., 2013) to recognize the markers and calculate the transformation matrix. For the speech input and audio guiding systems, we used PocketSphinx (Huggins-Daines et al., 2006) to process speech input and used a text to speech engine to generate the audio output.

4.2.4 Evaluation

The goal of Talkit is to help people with VIs understand 3D models quickly and naturally. We evaluated Talkit with target users, seeking their feedback to assess the overall user experience. We were also interested in whether they could effectively and efficiently find annotated elements and trigger the annotations.

We didn’t formally measure the accuracy of the recognition algorithms in Talkit in isolation, because they are heavily affected by other factors like the angle of the camera and the distance between the camera and the model. Instead, our study is designed to reflect how the system (including its algorithms) would perform in real use cases.

4.2.4.1 Method

Participants

We recruited eight legally blind participants (4 males, 4 females) whose ages ranged from 28 to 61 years (mean = 43.8, SD = 11.6). None of these participants had taken part in the exploratory study. Six participants identified as blind, while the remaining identified as low vision. Six participants had a college or graduate degree, and two participants had only graduated from high school. Six participants were familiar with braille, but only three read braille daily. All participants owned an iPhone. We compensated each participant 15 USD per hour and reimbursed transportation expenses up to 60 USD.

Apparatus and Materials

A researcher annotated four models from Thingiverse and printed them on a Makerbot 5th generation printer. As shown in Figure 4.9, the four models included:

1. *Cell*: A cell model modified from thing: 689381 (PaLEoS, 2015), the researcher annotated five elements on the model. Some elements have duplicates. For example, there are two "Lysosome." We only used this model for teaching and training purposes.
2. *Tank*: A tank model modified from thing: 90265 (xehpuk, 2013), the researcher marked five elements on the model. The model is symmetric, and there is "road wheel" on each side of the model.
3. *Terrain*: A terrain map (Figure 4.9c). We recorded labels for "Mountain," "River," and "Plain."

4. *Globe*: A globe model modified from thing: 17336 (bld, 2012), the researcher annotated five continents.
5. *Map*: A map model downloaded from OSM Buildings (2019). The researcher annotated five buildings on the model.

We printed a 3D tracker and prepared stickers. We used monochromatic filament and attached paper fiducial tags on the tracker. In practice, users with VIs could ask their friends for help when making a tracker or print it with a dual-extruder printer. The color paper stickers we used can be found in most crafts stores.

We ran Talkit on a 13-inch Macbook, and used its camera, microphone, and speaker as input and output devices. A researcher used another computer to remotely control the MacBook and record the entire study. We instrumented Talkit so that it logged interaction events such as speech commands and the

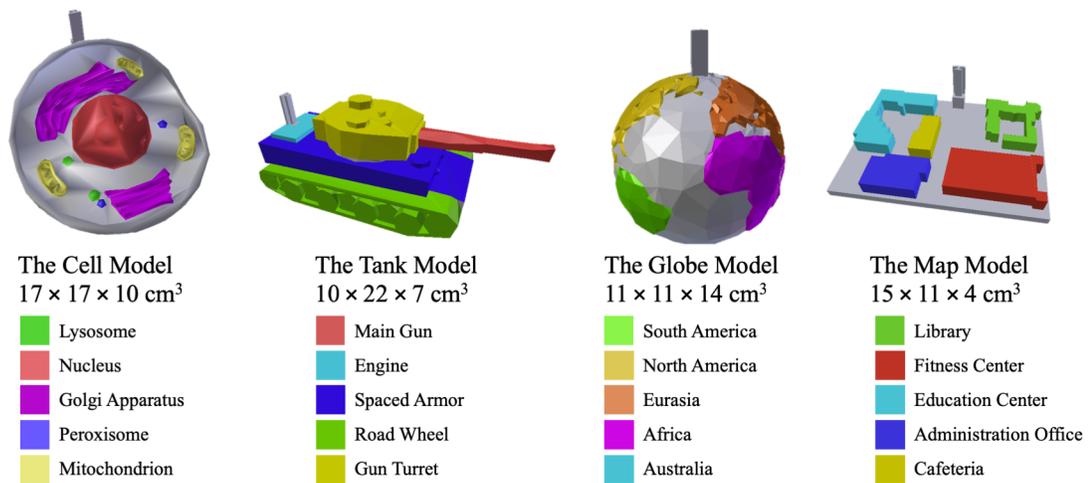


Figure 4.16: Four models used in the evaluation study of Talkit. The models and their elements are highlighted in color, and their names and dimensions are listed in the figure as well.

timestamps for each activated element.

Procedure

The study consisted of one session that was about 60 minutes long. After introducing the project to a participant, we (1) trained participants to use Talkit, (2) asked them to perform two tasks with the models, and (3) interviewed participants to gather subjective feedback.

After a 20-minute training session using the Cell model, we asked each participant to perform two tasks with the Globe, Tank, and Map models for a formal evaluation. We presented a participant with one model at a time, asking them to first explore the model without Talkit. We counterbalanced the order in which the models were presented with a Latin triangle.

Task 1. The goal of the first task was to see whether and how long it took participants to find all the annotations on a model. Participants were instructed to use Talkit to find and count all the annotations on a model. Participants began their exploration when we said “start.” When they thought they found all the annotations, they said “end.”

Task 2. The goal of the second task was to determine whether a participant could efficiently find a specific annotated element on a familiar model. We gave participants the option of exploring the model for up to two minutes until they felt comfortable with it. Then a researcher said the name of an element (e.g., “Engine” for the Tank model) and then said “start.” Participants used Talkit to find the element and trigger audio annotation and then said, “end.” Because we relied on participants’ confirmation, the time measured from this process will be slightly longer than what the participants actually spent. However, we think

the tolerance is acceptable as speaking a phrase takes less than a second and we would like to have participants confirm that they find the target element. We repeated this process for all five elements on each model, such that each element was activated once. We randomized the order of elements for each model. Eight participants completed a total of 120 trials for five elements on each of the three models.

At the end of the study, we conducted a short interview with a modified Standard Usability Scale (SUS) (Brooke, 1996) to collect subjective feedback from the participants. Our SUS consists of ten statements. We asked participants to rate the statements on a scale from 1 to 7, where 1 was strongly disagree and 7 was strongly agree. To reduce acquiescence and extreme response biases, we mixed positive and negative statements.

4.2.4.2 Results

The Time Needed to Access Annotations

In the first task, all participants found five elements on each model within three minutes (see Figure 4.17). Across all participants and models, P2 spent the longest amount of time (176 seconds) on the Tank model, while P7 spent the shortest amount of time (16 seconds) on the Map model. On average, participants spent 80 (SD = 46), 70 (SD = 40), and 43 (SD = 30) seconds respectively on the Tank, Globe, and Map models. Participants were generally less familiar with Tanks (especially P2 and P5), so the Tank model required more time to explore. Meanwhile, The Map model had a simple layout that lead to lower exploration times.

In the second task, participants located an element with a mean time of 7 seconds (SD = 8). Figure 4.18 shows the histogram of the time participants spent for this task. Participants spent less than five seconds on 62 out of 120 trials. The fastest 100 trials took less than ten seconds. Most participants memorized the

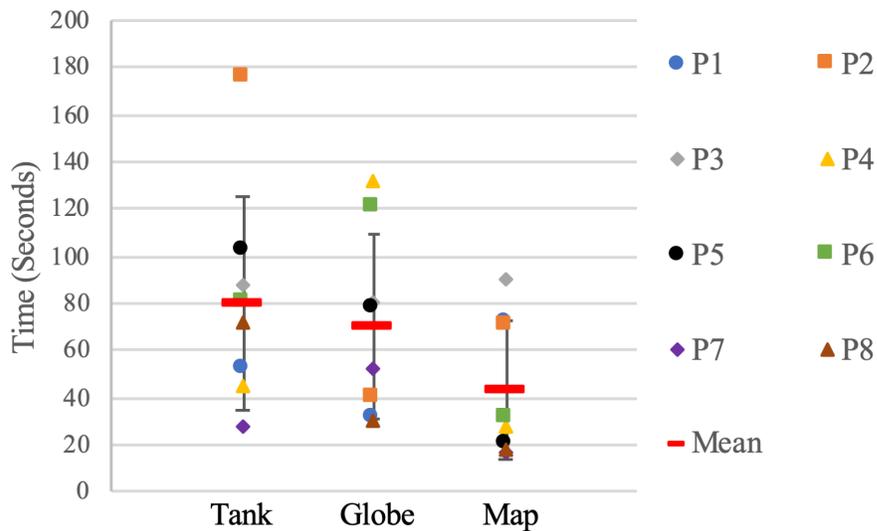


Figure 4.17: The time participants spent finding all five elements on each model using Talkit. Mean times for all participants are shown in red lines.

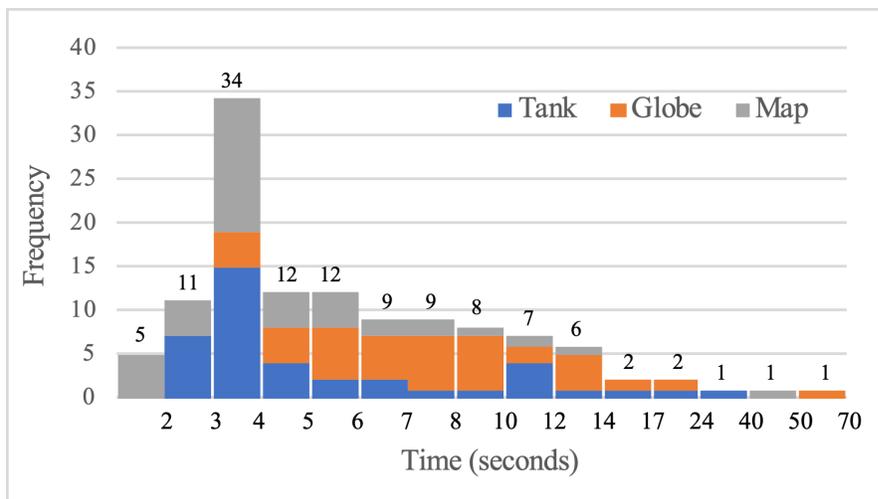


Figure 4.18: A histogram of the time participants took to locate an element, categorized and colored by models.

layout of the Map model and pointed to a target element in two seconds. They often needed to rotate the Globes and Tank models, sometimes using the audio guiding system to find the element, which took longer than finding elements on the Map model.

In a few cases, participants repeatedly explored certain areas that did not contain the target elements. For example, when asked to find “South America,” P6 believed the element was right below “North America,” and searched for it in the Pacific Ocean for over a minute. P3 did this as well when searching for the “Cafeteria” on the Map and “Spaced Armor” on the Tank.

Subjective Feedback

Overall, participants highly rated Talkit, and thought it was easy to learn and use, and helpful for understanding models. Table 4.1 shows responses to the modified SUS. Four participants said the most difficult part of the experience was positioning the model and their fingers (relatively negative scores for S3, S7, S8, S9, and S10). However, they felt they could learn how to do this better with the audio guiding system. P2 said it had a “slight learning curve,” which shared similarities with P4’s comment, “It just takes a bit of patience to get the finger placement correct.” Most participants thought they could handle the sticker and tracker by themselves, while P1 felt the tracker was difficult to place at the very beginning (a score of 3 for S2).

Participants expected Talkit to provide more information, and the models to be more available. They thought the system “has a lot of potential,” and encouraged us to add more detailed annotations, which led to lower scores for S6. P8 said, “I want to add information about what’s around the [annotation]

1	2	3	4	5	6	7	Mean
S1. I would like to use this system to learn three dimensional concepts.							6.5
				1	2	5	
S3. The system is easy to use							6.5
				1	2	5	
S5. I think the system would help me understand models.							6.875
					1	7	
S7. I would imagine that most people would learn to use this system very quickly.							6.375
					5	3	
S9. I felt very confident using the system.							6.375
					5	3	
S2. Its difficult to handle the tracker and the sticker by myself.							1.75
3	4	1					
S4. I would need the support of a technical person to be able to use this system.							1.25
6	2						
S6. I think the information you can get with this system is too limited.							2.375
1	4	2	1				
S8. The system is cumbersome to use.							1.625
4	3	1					
S10. I needed to learn a lot of things before I could get along with this system.							1.375
5	3						

Table 4.1: The ten statements (S1 S10) of the modified Standard Usability Scale (SUS) of Talkit’s evaluation study, with their histograms and means scores. Each statement was scored from 1 (stands for strongly disagree) to 7 (stands for strongly agree). For positive statements (S1, S3, S5, S7, S9), higher scores are better. For negative statements (S2, S4, S6, S8, S10), lower scores are better.

and how to find the [annotation] if I don't know where it is." In addition to more detailed information, two participants wanted 3D models to be more available. P8 gave a score of 5 for S1, explaining that "you need to have someone fabricate these models."

In the follow-up interviews, all participants indicated that they enjoyed using the system with its gesture and speech commands. P2 expressed her excitement, "I wish it was around when I was a kid." P8 said the most enjoyable part of this system is "the fact that you can move the models around" and "it's intuitive." The other participants (P1-P7) highlighted that, the most enjoyable part of this system is getting audio feedback from whatever I'm touching and "finding out what everything is."

The participants suggested making the system customizable. To avoid accidentally activating audio annotations, the current system required users to hold their finger over an element for 0.2 seconds to retrieve its annotation. P6, P7, and P8 commented on this feature, and thought we should enable users to customize the wait time. In addition to the delay, P7 envisioned a system that allowed users with VIs to annotate models themselves.

P3 and P8 suggested improvements to the audio feedback. They said sometimes it could be hard to find elements on an unfamiliar model, and the system only said "unmarked" to indicate there was no element under their fingers. This could be improved by providing additional guidance information such as the locations of nearby elements. This feature could prevent the cases where participants stuck to unmarked areas when looking for certain elements.

4.2.4.3 Discussion

Talkit achieved promising results in the user studies. Participants with VIs found five elements in an annotated model in 64 seconds (SD = 40) and found a specific element and triggered its audio annotation in 7 seconds (SD = 8) on average.

Talkit can support multiple interactive elements on an A3Ms. However, smaller elements would be more difficult to be selected. In the studies, participants could quickly locate small elements like “Main Gun,” which had a diameter less than 1 cm. The smallest recognizable element in Talkit depends on several factors like the user’s tolerance, the model design, and the accuracy of the tracking system. In the future, we will explore different tracking methods to improve the performance of the system.

The user input methods could be improved. In the next chapter, we will further discuss how to improve the interaction design of Talkit with the findings from a series of studies.

4.3 Summary

In this chapter, we introduced two sensing technologies for A3Ms through two systems.

Tickers and Talker is a toolkit that augments 3D models using acoustic sensing. We applied acoustic properties to create Tickers: small percussion instruments that generate unique sounds when strummed. Talker is a signal processing application that uses a microphone to sense and classify Ticker sounds, and speaks the label corresponding to the strummed Ticker. We evaluated the design of the final prototypes. Our system showed promising results as it achieved an overall mean accuracy of 93 percent in classifying activated labels. Overall, all participants with VIs reacted very positively to the system and were eager about the idea of using it. However, we found the performance of Tickers and Talker could be limited in noisy environment.

To make A3Ms more robust, we designed Talkit, a computer vision application. Makers (e.g., hobbyists, TVIs, and friends of people with VIs) can mark model elements and associate them with text annotations. A user with VIs can then print the augmented model, launch the Talkit application, and access the annotations by touching the model and following Talkit's verbal cues. Talkit uses an RGB camera and a microphone to sense users' inputs so it can run on a variety of devices. We evaluated Talkit with eight people with VIs. On average, with Talkit, blind people found a specified annotation on a model in an average of 7 seconds (SD = 8).

Different sensing technologies have their own advantages and disadvan-

tages. Acoustic sensing is simple to implement. However, due to the limited bandwidth of audio signals, the accuracy of this approach is likely to decrease when adding more interactive components on a model. Moreover, this method is sensitive to noise. Computer vision is more robust in noisy environment and can potentially have more interactive components. However, this method relies on a camera, and a user has to perform all interactions within the limited field of view of the camera. Thus, acoustic sensing is more suitable for models with a small number of interactive components, while computer vision has more advantages when used in noisy environment with complex models. We chose computer vision as the main sensing technology for other projects and studies in this dissertation considering its robustness.

These two systems received a lot of positive feedback from users, which encouraged us to further explore other research questions associated with A3Ms. For example, how should we design the content of A3Ms? What design tools will help people design A3Ms? These questions were partially touched in this chapter, and we will explore them in more rigorous and comprehensive studies in the following chapters.

CHAPTER 5

DESIGN STUDIES

In Chapter 4, we explored different sensing technologies and chose computer vision as the main technology for A3Ms. In this chapter, we will investigate the interaction design of A3Ms in a series of studies. While some content is built on computer vision-based A3Ms, most of the design guidelines and insights we derived from the studies can be applied in other types of A3Ms.

This chapter will start with an elicitation study with 12 participants with VIs. In this study, we aim to (1) understand how people with VIs explore tactile models and (2) elicit the input techniques of A3Ms from the perspectives of people with VIs. In the first section of the study, we observed participants' behavior as they explored and identified models and their components. In the second section, we elicited user-defined input techniques that would trigger various functions from an interactive model. Through this study, we identified common exploration behavior and strategies and aggregate their elicited interaction techniques.

In addition to the elicitation study, we hosted two workshops with TVIs to understand how an A3M should be designed to serve as an effective teaching aid. In the workshop, 35 TVIs modified 3D models that researchers prepared and added interactive elements to the models. TVIs are domain experts in special education, and they provided insights about how to adapt A3Ms to classrooms during the workshops.

With the findings from these studies, we iterated the design of A3Ms and developed Talkit++, an iOS application that supports the use of A3Ms. To further

test the application and understand how A3Ms work in the field, we conducted a co-design study with three TVIs, who deployed A3Ms in their classrooms.

At the end of the chapter, we conclude with design guidelines that we derived from these studies.

5.1 The Elicitation Study

5.1.1 Introduction

Augmented 3D printed models draw a lot of attentions from research community. Researchers have developed different sensing technologies to augment printed models. However, prior work focused on the technical aspects of sensing input on the models and producing output; no work has investigated the methods and behaviors of people with VIs as they explore tactile 3D models and determined their needs for interactivity. To date, most A3Ms supported straightforward interactions designed by researchers (e.g., pointing to a continent on a globe model to get its label). These designs didn't fully explore the design space and power of A3Ms.

In this section, we aim to understand people with VIs' needs and preferences for the design of interactivity in 3D printed models. Specifically, we investigate two research questions:

RQ1: How do people with VIs explore tactile models (that are not interactive)?

RQ2: What interaction techniques are effective in A3Ms?

Answering both questions is critical to our aim. A3Ms are a new medium that present undiscovered design opportunities. Unlike designing interactions for traditional web browsers (Morris, 2012) or desktop systems (Wobbrock et al., 2009) that have established functionalities, interaction design for 3D printed models is not established and we do not yet know what functions are needed or useful. The goal of interactive systems is to support and adapt to users' behaviors (Good et al., 1984), rather than the opposite. Thus, RQ1 contributes to the fundamentals of A3Ms, while RQ2 takes a direct action to seek interaction techniques for A3Ms.

To answer the questions, we conducted a two-section study with 12 legally blind participants. In the Exploration section of the study, we observed participants as they performed several tasks to explore and identify 3D printed models and their components (RQ1). In the Elicitation section of the study, we elicited interactions from participants, asking them to define input techniques for five interactive functions for an A3M (RQ2). The term "elicitation" was introduced in prior work by Wobbrock et al. (2009), and the design of our Elicitation section is similar to the studies in prior work (Wobbrock et al., 2009; Morris, 2012).

Our findings reveal that 3D model exploration involves five major types of exploration activities (e.g., counting similar tactile elements), four hand postures (e.g., using one hand to hold a model on the table), and eight gestures (as shown in Figure 5.1). In the second section of the study, participants suggested touch gestures on the model, speech commands, and buttons to trigger interactive events.

In this section, we will start with the study design. Then, we will present the results from the Exploration section and the Elicitation section separately. We



Figure 5.1: The eight gestures identified in the Elicitation section of the study. The movements of dynamic gestures are marked with arrows. We highlight the hands that were performing the gestures and mark the remaining parts in grey.

conclude with a set of recommended interactions and functions.

5.1.2 Study Design

5.1.2.1 Study Objectives

We aim to draw design implications from (1) people with VIs' exploration behaviors of 3D printed models and (2) user-defined input techniques. Thus, we conducted a study to answer the two research questions posed in Section 5.1.1. The study consists of two parts: the Exploration section and Elicitation section, which are designed to answer the research questions respectively.

5.1.2.2 Participants

We recruited 12 legally blind participants (4 males, 8 females) whose ages ranged from 23 to 60 years (mean = 40.75, SD = 13.15). Eleven participants identified as blind, while the remaining one identified as low vision. Eight participants had a college or graduate degree, and four participants had only graduated from high school. Eleven participants were familiar with braille, but four of them didn't read it regularly. Eleven participants had iPhones, while the remaining one did not use a smartphone. We compensated each participant 15 USD per hour and reimbursed transportation expenses up to 60 USD.

5.1.2.3 Apparatus and Materials

We printed three models for the study. These models are used in both the Exploration and the Elicitation sections. We downloaded three models from Thingiverse (2019) and OSM Buildings (2019). We chose these models because they are common in education and navigation tasks. We anticipated the target users of A3Ms will use similar types of models in the future. Since most interactive models require auxiliary components (e.g., the 3D trackers in Section 4.2 to sense user's touches, we added a stick to the top of each model to represent an auxiliary component, as shown in Figure 5.2. The models were printed on a Makerbot 5th generation printer. The three models included:

1. *Cell*: A cell model modified from thing: 689381 (PaLEoS, 2015). The model has a Nucleus, two Golgi Apparatuses, three Mitochondrion, and four semi-spheres representing Peroxisomes.



Figure 5.2: Three printed models used in the study. From left to right: the Cell model, the Globe model, and the Map model.

2. *Globe*: A globe model modified from thing: 17336 (bld, 2012). The model has seven continents.
3. *Map*: A map model downloaded from OSM Buildings (2019). The model includes several buildings.

For the Elicitation section, we developed a Wizard-of-Oz prototype. When participants perform an input technique, we play corresponding audio feedback to simulate output. Thus, we developed a Python script that used the text-to-speech engine in a Macbook to speak texts.

We used two cameras to record the study: a fixed camera mounted on the table and a hand-held camera operated by a researcher.

5.1.2.4 Procedure

The study consisted of one session that was about 60 minutes long. The session included two sections, which we refer to as Exploration and Elicitation.

After introducing the project to a participant, we began the Exploration section, in which we asked them to explore three models by performing three tasks. Then, we began the Elicitation section. In a typical elicitation study, researchers prompt a user with the effects of an action, known as referents, and the user is expected to provide the causes of the action, known as signs (Good et al., 1984). In our study, we asked users to define signs (input techniques) for six Referents (functions) with three models.

We designed tasks for the Exploration section that would prompt participants to both explore the models as a whole and examine smaller model components. Hatwell (1987) and Lederman and Klatzky (1993) found that exploration using tactile perception involves two stages: an overview and detailed exploration; participants would stop exploring at the first stage if no further instruction was given. Thus, our three tasks were:

Task 1: *Identify the Model.* We asked participants to tell us what the model represented. If a participant didn't know what the model was, the researchers told her.

Task 2: *Describe the Shape of an Element.* We asked participants to describe a designated element on each model: Nucleus in the Cell model, North America in the Globe model, and the round building in the Map model. If a participant couldn't find the specific element, the researchers showed her the location of the element.

Task 3: *Describe the Shapes of Nearby Elements.* After Task 2, we asked the participant to explore and describe the elements near the designated element from Task 2.

When performing the tasks, we asked participants to think aloud and explain what they were feeling and doing. If they asked questions about the content on the models, we would answer them after they finished the tasks. We gave the participant one model at a time and counterbalanced the order in which the models were presented with a Latin triangle. The Exploration section lasted around 20 minutes, and participants were encouraged to take a break after finishing each model.

During the Elicitation section, we invited participants to design input techniques they wanted to use to get audio output from A3Ms. We used the same models we used during Exploration and asked each participant to define input techniques for six functions, henceforth referred to as referents. The first four referents were demonstrated in interactive printed maps (Götzelmann, 2016; Taylor et al., 2016). The last two referents were inspired by the study in Section 4.1.5. All referents used audio as feedback. These referents included the following, with examples from a globe model:

Referent 1: *Get General Model Information*. The system provides a high-level description of a model: “this is a globe with seven continents.”

Referent 2: *Select an Element and Get its Name*. The user selects a continent and the system speaks its name: “North America.”

Referent 3: *Select a Sub-Area of an Element and Get its Name*. The user selects an area on North America and the system says: “This is where the Rocky Mountains are.”

Referent 4: *Get More Information*. The system provides information about the Rocky Mountains from Wikipedia.

Referent 5: *Record Notes*. The user performs a technique to record notes about the Rocky Mountains.

Referent 6: *Retrieve Notes*. The system speaks the previously recorded notes about the Rocky Mountains.

As is standard with elicitation studies (Wobbrock et al., 2009), we went through each referent with a participant and asked her to elicit input with the help of the Wizard-of-Oz audio prototype. We prompted her by saying “If the models can sense your behaviors and understand your speech, how would you like to interact with the models?” Then, a researcher demonstrated the audio output for each referent. For example, when demonstrating Referent 1 for the Globe model, the researchers used a script to produce synthesized speech for the sentence: “This is a globe model. You can find Asia, North America, South America, and Africa in this model.” After demonstrating all referents, the participant was asked to design input techniques to trigger these referents. When the user performed a possible input technique, the researcher played the audio output to simulate the complete interaction.

For each referent and each model, participants were asked to define one input technique. They could use the same inputs for the three models. If they felt that one of the referents was unnecessary for a given model, we allowed them to skip this referent. As with the Exploration section, we gave each participant one model at a time, and counterbalanced the order in which the models were presented with a Latin triangle.

5.1.2.5 Analysis

Two researchers analyzed the videos recorded by two cameras. We analyzed the video data from the Exploration section in three stages, following Lederman et al.'s video coding method (Lederman and Klatzky, 1987). This method allowed us to combine data from our observations of participants' behavior and their reported thought process. In the first stage, we carefully reviewed the videos from the hand-held camera, and coded them using digital note cards. A note card contained a static frame from the video, an identification code, and text explaining this frame and related dialog from the video. A researcher initiated the transcription process by creating a series of note cards for a video clip. Then the other researcher validated the note cards to make sure the cards represented the video clip.

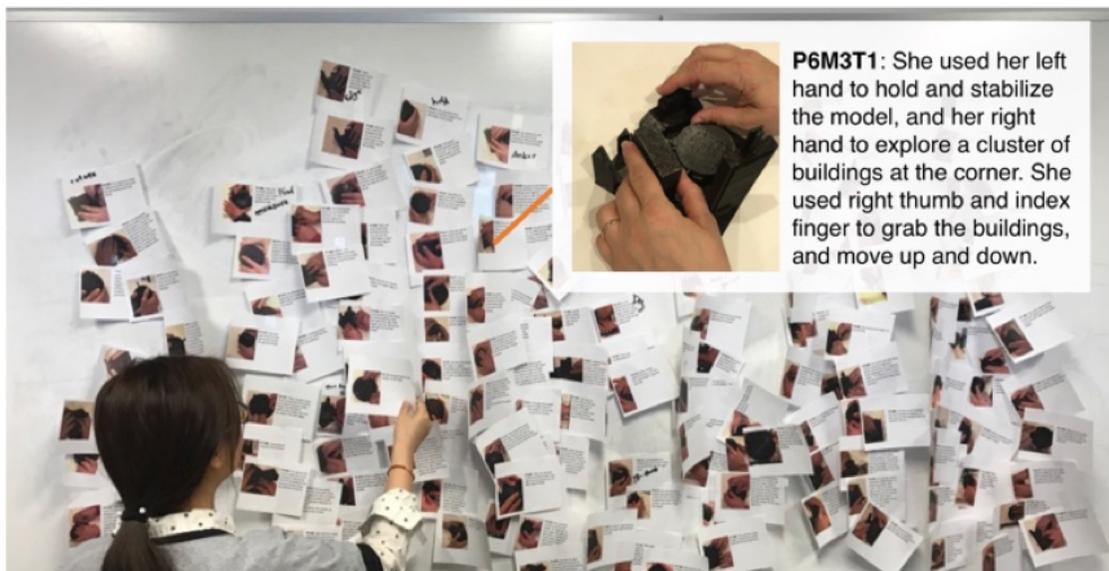


Figure 5.3: A researcher was clustering the note cards. Another participating researcher took the photo.

In the second and third stages, two researchers categorized the note cards into themes in an iterative process. In the second stage, the researchers printed the note cards, and clustered them into several groups based on the performed hand postures and gestures described in the note cards, as shown in Figure 5.3. Then, they identified themes, like common exploration activities and questions participants asked about the models. In the third stage, the researchers independently reviewed each note card with the identified themes and validated the themes. We collected and coded a total of 300 note cards.

For data from the Elicitation section, we sorted suggested input techniques, and transcribed only the audio data of the videotapes. We calculated the Max-Consensus and Consensus-Distinct Ratio for each Referent, following Morris et al.'s approach (Morris, 2012). We developed themes from the transcription using axial coding (Saldana, 2015).

5.1.3 Findings: Exploration

We describe the findings from the Exploration section of our study. We found that participants performed five distinct exploration activities, frequently used four hand postures, and performed eight gestures during the exploration tasks.

5.1.3.1 Exploration Activities

When a participant explored a model, she sensed the texture and shape of the model, measured the size of the elements, counted the number of elements, compared similar elements, and communicated with the researcher. In the study, these activities appeared in all three tasks, including Task 2 and 3 where

participants could complete the tasks by only sensing the texture of a model and describing the shapes of its elements. Thus, all these five activities are important in tactile exploration.

Sensing

Sensing the shape and texture of a model was the most common activity participants performed when exploring the models. Most of the time participants moved their fingertips back and forth against the model to sense its texture; they occasionally used their palms and fingernails as well. To feel the shape of a model, they performed static gestures like holding the model firmly in their hands.

Measuring

Five participants used their hands to approximately measure the sizes of components on the models. Participants measured components on the Map model most, feeling the heights and widths of the rectangular buildings and the radius of the round building using proprioception. Participants performed these “measuring” activities during all tasks and for all models.

Comparing

Four participants mentioned they compared the shapes of two elements to confirm whether they represented the same concept. For example, P8 and P11 touched two semi-spheres on the Cell model to check whether they both represented Peroxisomes.

Counting

When a specific shape appeared more than twice, many participants counted the instances of the shape. We noted a counting activity whenever a participant verbally counted similar shapes and found seven participants performed counting activities when exploring the Cell and Map models across three tasks. On the Cell model, participants counted the holes in the Nucleus. On the Map model, they counted the buildings and components of the buildings (e.g., the entrances of the round building).

Communicating

Eight participants communicated with the researchers during the three tasks. Their communication generally served three purposes:

- *Indication:* They used different gestures to tell the researchers that they found the specified element in Task 2.
- *Explanation:* They described their behavior and pointed out certain elements to the researchers.
- *Inquiry:* They asked questions about the identities of elements during the three tasks.

5.1.3.2 Hand Postures

Identified Postures

Participants held and manipulated the models using four distinct postures. Sometimes they only used one hand to explore the model, leaving the other

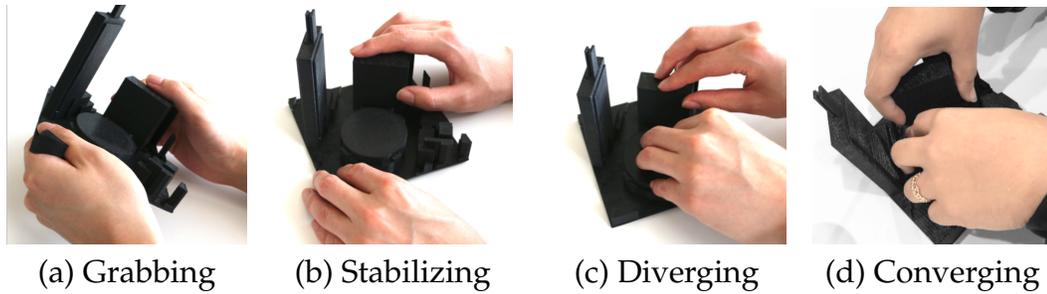


Figure 5.4: Four identified hand postures from the Exploration section.

hand to hold or stabilize the model. In other cases, they used both hands to explore the model. As shown in Figure 5.4, we identified four postures: *Grabbing*, *Stabilizing*, *Diverging*, and *Converging*. We described the four postures and their advantages in Table 5.1. All participants used the four postures throughout the exploration tasks except for P12, who only used the Grabbing posture.

Some of these findings are consistent with the kinematic chain model proposed by Guiard (1987). The kinematic chain model describes the interactions between dominant hand and non-dominant hand and provides a theoretical framework for bimanual actions. The model suggests that dominant hand and non-dominant hand work cooperatively but asymmetrically. The non-dominant hand tends to act first and set a reference frame for the other hand. The dominant hand tends to act later and operate within the reference frame. In our study, the four identified hand postures follow the kinematic chain model. For example, the Grabbing posture and the Stabilizing posture involve a non-dominant hand that sets the reference frame for a model (e.g., holding or stabilizing a model) and a dominant hand that explore the model.

	Posture	Description	Advantages
Exploring Using One Hand	Grabbing	Use one hand to hold the model in the air and perceive tactile information from the other hand.	Allowing a participant to use one hand to navigate and rotate the model freely.
	Stabilizing	Use one hand to fix a model on the table, and the other hand to learn the model.	Not only enabling a participant to explore the model stably, but also providing a reference point from his fixed hand.
Exploring Using Two Hand	Diverging	Put two hands on two elements separately, and each hand could perform different gestures.	Enabling a participant to explore two elements at the same time.
	Converging	Use both hands to explore a single element.	Allowing a participant to explore large elements (e.g., the Golgi Apparatuses in the Cell model) using both hands.

Table 5.1: The four identified hand postures.

Patterns of Hand Postures

We found participants actively and unconsciously switched among postures to explore the models. The posture a participant adopted depended on (1) the shape of the model being explored, (2) the size of the area the participant was perceiving, and (3) the exploration activity performed.

The shape of a model afforded the postures participants used to explore it. For example, ten participants only used the Grabbing posture when exploring the Globe model. P5 reflected on his posture and said, “I think it’s easier with my hands [to hold the Globe model in the air].” While P6 thought she didn’t use the Grabbing posture on the Map model because “I don’t feel comfortable holding it.” On the other hand, participants were more likely to put stable mod-

els (e.g., the Cell and the Map models) on the table and adopt the Stabilizing, the Diverging, and the Converging postures. P1 said he put the Map model on the table because “it has a solid base.” P7 said she would have put the Globe model on the table if it had “the shape of a teardrop.”

The size of the area participants explored also influenced their hand postures. For example, participants tended to use the Diverging posture when exploring large elements (e.g., the Golgi Apparatuses on the Cell model).

Postures also varied depending on what exploration activity the participant was performing. For example, all comparison activities were associated with the Diverging posture, with which participants could examine two elements at a same time. Meanwhile, in all communication activities participants used the Grabbing and the Stabilizing postures. Participants used one hand to grab or stabilize the models while using the other hand to point out elements to the researchers.

5.1.3.3 Gestures

Identified Gestures

We found eight gestures in the study. We classified a gesture by which parts of the hands were involved and whether it was static or dynamic. We define a static gesture as a gesture in which a participant pauses in the middle of performing an action with her hand. By contrast, we define a dynamic gesture as a gesture in which a participant performs an action that involves continuous movement. Static gestures were associated with cutaneous perception, while dynamic gestures involved haptic perception as well. In the study, participants

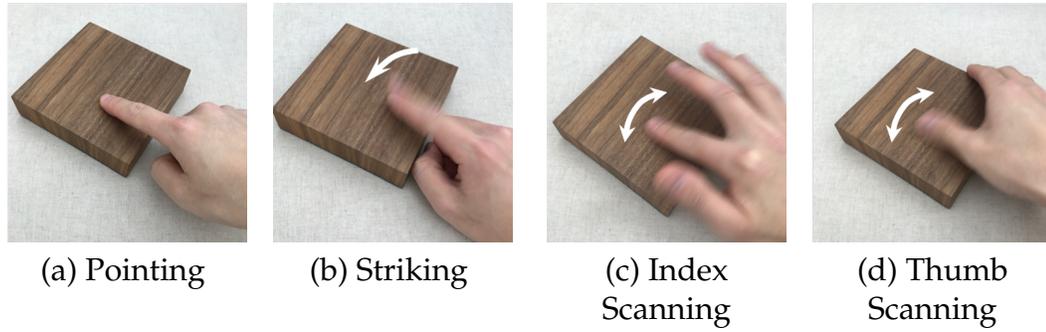


Figure 5.5: Four identified single-finger gestures from the Exploration section.

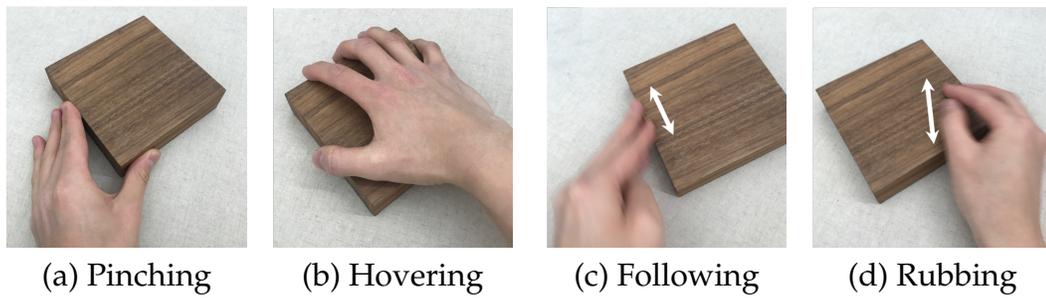


Figure 5.6: Four identified multi-finger gestures from the Exploration section.

performed four single-finger gestures (i.e., *Pointing*, *Striking*, *Index Scanning*, and *Thumb Scanning*, as shown in Figure 5.5) and four multi-finger gestures (i.e., *Pinching*, *Hovering*, *Following*, and *Rubbing*, as shown in Figure 5.6). The *Pointing*, *Pinching*, and *Hovering* gestures were static gestures, while the others were dynamic. We described the gestures in Table 5.2.

Patterns of Gestures

Participants unconsciously chose gestures based on (1) the ongoing activities and (2) properties of the areas they were exploring.

Participants used different gestures for different exploration activities. Table 5.3 shows the Gesture-Activity matrix, where we marked all combinations that

	Posture	Description	Movement	Parts of Hands
Single-finger Gestures	Pointing	Hold an index finger on a model. Five participants used this Pointing gesture.	Static	Index fingers
	Striking	Strike a point on a model with quick light blows. Six participants performed this gesture.	Dynamic	Index fingers
	Index Scanning	Use index fingers to scan an area. Nine participants did this gesture.	Dynamic	Index fingers
	Thumb Scanning	Use thumbs to scan an area. Four participants did this gesture.	Dynamic	Thumbs
Multi-finger Gestures	Pinching	Grip a model using thumb, index, and middle fingers, and hold this gesture for a while to perceive tactile information. Five participants did this gesture.	Static	Thumbs, index, and middle fingers mainly
	Hovering	Put an entire hand on a model without observable movement. Eight participants used this gesture.	Static	Fingertips, palms, and finger pads
	Following	Use more than two fingers together to follow the edge of an element. Ten participants used this gesture.	Dynamic	Fingertips and finger pads
	Rubbing	Use nails, fingertips and finger pads to move around an area. All participants used this gesture.	Dynamic	Nails, fingertips and finger pads

Table 5.2: The eight gestures identified in the Exploration section.

	Sensing	Measuring	Comparing	Counting	Communicating
Pointing					X
Striking					X
Index Scanning	X	X	X	X	
Thumb Scanning	X				
Pinching	X	X			
Hovering	X	X		X	
Following	X				X
Rubbing	X	X	X	X	X

Table 5.3: The Gesture-Activity matrix. We marked all the combinations of exploration activities and gestures that appeared in the Exploration section of the study.

appeared in the study. While the Rub gesture was more capable and could support all five activities, the other gestures served fewer activities. Three gestures only served a single type of activity.

The Pointing and the Striking gestures were only used when participants communicated with the researchers. Participants only used the Thumb Scanning gesture to sense the texture and shapes of elements of a model.

In addition to the influence of the activities, participants also chose gestures based on the complexity and the sizes of the areas they were exploring. When a model was simpler, they tended to use fewer fingers with gestures like Index Scanning, Thumb Scanning, and Hovering. P2 said, “If it’s a larger, general model, I would use even the palms of my hands I want to get everything at once, when I want to be specific, I will revert to my pointers.” Other participants also shared similar thoughts. This is because fingers, especially fingertips, have a greater density of receptors, and are more sensitive to detailed tactile information (Hatwell et al., 2003). On the other hand, the palms have larger contact surfaces and can sense larger elements or an entire model at the same time.

5.1.4 Findings: Elicitation

Ten of the 12 participants designed input techniques for all referents. Most used the same input techniques for referents across the three models. P5 thought recording notes and retrieving notes were unnecessary, and P12 thought selecting an area was not a reasonable referent. We report the input techniques participants proposed and summarized patterns from their designs.

5.1.4.1 User-Defined Input Modalities

In total, participants proposed 82 interactions for six referents, 58 of which were distinct on a per-referent basis. The suggested interactions covered three input modalities: gestures, speech input, and buttons. In Table 5.4, we divided the interactions into three modalities. Among all user-defined interactions, gestures were the most common modality and had the biggest variations.

Gestures

All participants except for P12 wanted to perform gestures on the models to trigger some referents. Some gestures they mentioned are commonly available on touchscreens or supported by screen reader software, such as tapping, touching, pressing with force, holding for a while and swiping. Some gestures had

	Total Techniques	Distinct Techniques
Gestures	56 (68.3%)	40 (70.0%)
Speech Input	13 (15.9%)	9 (15.5%)
Buttons	13 (15.9%)	9 (15.5%)
Total	82	58

Table 5.4: The total and distinct Techniques broken down by different interaction modalities.

variations. For example, participants brought up different tapping gestures like a single-finger single tap, a two-finger single tap, and a two-finger triple tap. Participants sometimes combined two gestures to create a new gesture (e.g., a tapping gesture followed by a press gesture).

Several participants designed gestures that were not based on current eyes-free touchscreen interaction. P2 wanted to shake a model to get general information about it and squeeze an element to get its name. P10 used her fingers to follow the edge of an area, which we defined as the Following gesture (see above), to select it. P9 scratched an area to select it.

Two participants described principles that guided their proposed gestures. P6 used speed as a parameter to trigger different referents. For example, a user could perform a swiping gesture quickly on an element to get general information about it, and then slow down to retrieve more detailed information. P8 introduced the Rochester Method (Scouten, 1967), also known as Visible English. He drew a symbol representing a question mark (as defined by the Rochester Method) to retrieve general information about a model. In prior work, Geißler (1995) proposed how to combine gestures with icons to create more complicated interactions, which is similar to some proposed gestures. For example, a single tap on an element (or an icon) will retrieve its general information while drawing a stroke of a question mark on the element will query the detailed information of the element.

Speech Input

Five participants mentioned speech input. They used simple commands such as “what is it?” “more information,” and “save information.” P11 said these

commands were inspired by Siri on the iPhone.

Buttons

Participants designed buttons to fulfill different functions and arranged them in different locations. For example, P5, P7, P12 wanted to have a button that could turn on the audio feedback of an A3M and speak general information about the model. P3 designed buttons to get detailed information and take notes. However, three participants emphasized that the buttons should not be incorporated into the models. P7 commented on the effect of buttons on the tactile information represented on a model, saying that “adding buttons to the structure almost takes away from actually teaching someone what is the thing you are touching.” They suggested placing the buttons at the bottom of a model or on an auxiliary component like the long stick we added.

5.1.4.2 Elicited Interactions for Referents

On average, each referent got 13.7 input techniques, and 9.7 distinct input techniques. When broken down into the three input modalities (gestures, speech, and buttons), there were 6.7 distinct gestures, 1.5 distinct speech input commands, and 1.5 distinct button arrangements per referent.

To further understand user agreement of proposed interactions for each referent, we calculated the max-consensus (MC) and consensus-distinct ratio (CDR). MC is the percent of participants who suggested the most popular input technique for a referent or referent/input modality combination. A higher MC value indicates more user agreement on the most popular user-defined input technique. For a referent or referent/interaction technique combination, CDR is

the percent of the distinct techniques that achieved a given consensus threshold among participants. This metric indicates the diversity of techniques that fulfill a certain user agreement. We used a threshold of two in the analysis.

In Table 5.5, we list MC, CDR, and the distinct input techniques that were above the consensus threshold for each referent. Across all referents, gestures received more user agreement with a mean MC of 20.8 percent, compared to speech input (mean MC = 11.1 percent) and buttons (mean MC = 12.5 percent). However, speech input and buttons were more favored for referents that do not involve specific model elements. For example, the referent “Get General Model Information” had a MC of 25 percent for buttons, but a MC of 17 percent for gestures. The referent “Get More Information” had a MC of 25 percent for speech input, compared to a MC of 17 percent for gestures. P3, who designed a button for “Get General Model Information,” said that “it doesn’t matter if you are feeling it (the model).”

For each referent, we can choose the technique with the highest number of the supporting participants (i.e., the first action in each row) as its elicited interaction. In this study, our elicited interaction set is conflict-free. Prior work introduced a method to eliminate conflicts in elicited interactions (Wobbrock et al., 2009).

5.1.5 Proposed Design

We propose a set of recommended interactions and functions. Our proposed design uses an RGB camera, a microphone, and a touchscreen from a tablet as sensors, and is an extension of our work in the previous chapter. With these

Referent	MC/CDR Type	MC	CDR	Above Threshold Distinct input actions	#
Get General Model Information	Overall	25%	23%	<ul style="list-style-type: none"> • Push a button on the long stick to turn on the model • Use index finger to press the model at beginning • Use index finger to touch the model at beginning 	3
	Gestures	17%	22%		2
	Speech	8%	0%		2
	Buttons	25%	50%		2
Select an Element and Get its Name	Overall	25%	40%	<ul style="list-style-type: none"> • Use index finger to tap once on the element • Use index finger to press the element • Use index finger to touch the element • Use index finger to tap twice on the element 	3
	Gestures	25%	44%		3
	Speech	-	-		2
	Buttons	8%	0%		2
Select a Sub-Area of an Element and Get its Name	Overall	33%	29%	<ul style="list-style-type: none"> • Use index finger to swipe on the area • Push a button on the long stick to switch the level of information 	4
	Gestures	33%	17%		2
	Speech	-	-		2
	Buttons	17%	100%		2
Get More Information	Overall	25%	40%	<ul style="list-style-type: none"> • “more information” • Use index finger to swipe on the element/area • Use index finger to tap twice on the element/area • Use index finger to tap on the element/area and hold 	3
	Gestures	17%	38%		2
	Speech	25%	100%		2
	Buttons	8%	0%		2
Record Notes	Overall	17%	22%	<ul style="list-style-type: none"> • “record notes” • Use index finger to tap on the element/area twice 	2
	Gestures	17%	20%		2
	Speech	17%	50%		2
	Buttons	8%	0%		2
Retrieve notes	Overall	17%	22%	<ul style="list-style-type: none"> • “retrieve notes” • Use index finger to tap on the element/area twice 	2
	Gestures	17%	20%		2
	Speech	17%	50%		2
	Buttons	8%	0%		2

Table 5.5: Six referents and their max-consensus (MC) values, consensus-distinct ratios (CDRs), and distinct input actions that were above a consensus threshold of two. The MC and CDRs were also broken down into three input modalities. Two referents didn't get any speech interaction. For each distinct input action, we list the number of the participants reporting in the “#” column.

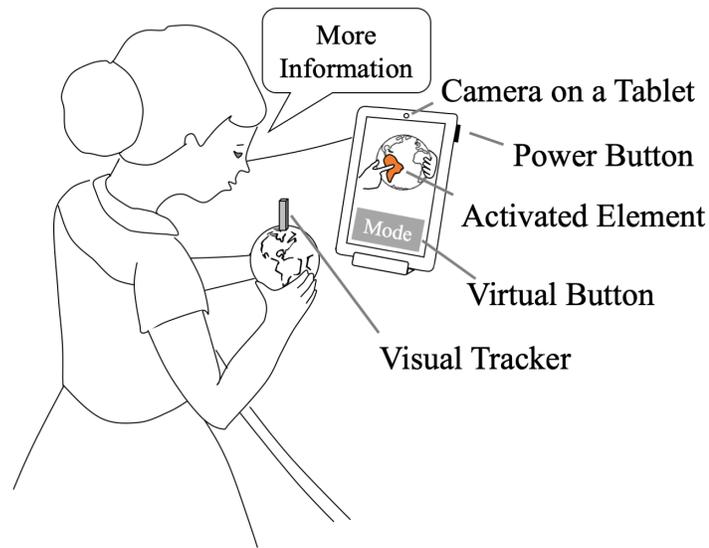


Figure 5.7: A use case for the proposed design. A user selects an element with her index finger and says “more information” to retrieve further information about the element.

Functions	Interactions
Turn on/off	Power button
Switch modes	Virtual button
Select a Sub-Area of an Element and Get its Name	Use index finger to tap once on the element/area
Compare two elements/areas	“Comparison” + Use index finger to tap once on two elements/areas
Get dimension	“Dimension” after selecting
Get more information	“More information” after selecting

Table 5.6: The functions and interactions for the proposed design.

sensors, a user can use gestures, speech input, and physical and virtual buttons to interface with an A3M, as shown in Figure 5.7 . We describe the functions and interactions of the proposed design in Table 5.6. There are six functions, which can be activated through buttons, gestures, and speech commands. The “switch modes” function is designed to handle different levels of information. For example, the Globe model may have one mode describing the names of continents and another mode presenting the information of countries.

The findings from both sections of the study contributed to our proposed design. The exploration behaviors we identified implied the potential functions blind people need. For example, the proposed design enables them to compare elements or areas and inquire the dimension information of an element. The elicited input techniques yielded insights about blind participants' preferences for different interaction techniques, as well as their opinions about A3Ms. For instance, we design virtual and physical buttons that enable users to control information flow.

5.1.6 Discussion

We conducted an elicitation study with blind participants and discussed design implications for A3Ms. In addition to A3Ms, the findings from our study can also contribute to the design of passive printed models. Passive printed models also faced challenges in terms of design and technology. Thus, model designers should consider our guidelines to improve tactile information for passive printed models. Moreover, designers should provide accessible information (e.g., tactile Braille prints, audio books) to explain the information included in the models to avoid confusion, if that information is not accessible interactively.

In the Exploration section of our study, we developed a new taxonomy to analyze exploration behaviors. We classified each note card in three dimensions: exploration activities, hand postures, and gestures. We defined hand postures based on the functionalities of hands and specified each gesture by its movement and which parts of the hands it involved. With this taxonomy, we coded all note cards. While prior work on tactile perception (Lederman and Klatzky, 1987) and elicitation (Morris, 2012; Wobbrock et al., 2009) introduced several

taxonomies that might yield other perspectives, we believe our inductive approach captured a range of salient behaviors in the relatively unexplored space of the perception of 3D models.

In the Elicitation section of our study, we asked participants to design input techniques to retrieve audio information about models and their components. We used audio as the default output since all prior work on A3Ms provided audio feedback. However, other output modalities, such as smells and vibrations, can also be applied to A3Ms. Visual feedback can also be important for people with low vision, who comprise the majority of the population of people with visual impairments (Szpiro et al., 2016). Future research can explore potential output techniques.

One limitation of elicitation studies is that participants are usually biased by their experience with prior technologies (i.e., legacy bias (Morris et al., 2014)). Thus, participants may not be able to propose new interaction techniques. In our study, researchers acted as the lead designers who wanted to use mainstream input technologies (e.g., buttons and gestures). The participants helped the designers materialize appropriate designs (e.g., how to use buttons to control A3Ms). Thus, the legacy bias did not affect the goals of the study. However, future research on A3Ms and gesture elicitation should be careful about the legacy bias.

We proposed a design of A3Ms using the sensors in a tablet. The sensors a designer chooses will affect the design of A3Ms. For example, RGB cameras are ubiquitous, but they are less capable of differentiating touching and pressing gestures. Moreover, sensors can also affect the design of auxiliary components. In the study, we added a long stick on each model to represent an auxiliary com-

ponent. We found that such components wouldn't bother users if they were designed to support users' exploration behaviors. Model designers should choose sensors based on use cases and adopt the findings from this study.

The proposed design enables a user with VIs to retrieve information from A3Ms, but we think A3Ms themselves could take more initiative and guide users through the exploration process of a model. Prior work suggested that by following some exploration patterns, users with VIs could learn tactile information more efficiently (Morash et al., 2014). A3Ms should be able to help users find the best strategy to explore models. However, we need further research to design accessible and efficient instructions to guide this process.

The findings of this study could be more substantial with future research. The identified exploration behaviors should be confirmed with more participants, and the design implications we concluded should be further examined. Moreover, the proposed design should be validated in an iterative design process.

5.2 The Workshop Study

5.2.1 Introduction

The elicitation study in Section 5.1 revealed how people with VIs would design A3Ms, especially the input techniques of A3Ms. However, people with VIs are not the only stakeholder of A3Ms. As teaching aids, A3Ms also play an important role in classrooms and are highly related to education practitioners like TVIs. Yet, little work has explored how to design A3Ms in an educational setting.

The gap between A3Ms and educational practice led to a series of unsolved research questions. How should we design and modify the physical model of an A3M to meet teaching needs? What interactive elements (e.g., audio labels) should we add to an A3M? How should researchers support the design and use of A3Ms in a teaching session?

In this section, we present a workshop study that investigates how to design A3Ms as effective teaching aids. We led two design workshops with 35 TVIs, domain experts who design tactile teaching aids for their students. The TVIs modified 3D models we prepared and added interactive elements on the models. Through the workshops, the TVIs provided suggestions about model modification (e.g., the size of physical models should be bigger), interactive elements (e.g., adding visual animations for low vision students), interaction techniques (e.g., supporting speech input), and pedagogical considerations (e.g., providing an overview before details).

The TVIs also provided feedback for the existing A3M system. During the workshops, they explored existing A3Ms using Talkit, an enabling desktop application demonstrated in Section 4.2. In the next section, we will describe how we developed Talkit++ with the feedback. The Talkit++ application was further examined in a co-design study.

In this section, we will report the study design and the results from the workshop study in details.

5.2.2 Study Design

The primary goal of the workshop study was to understand how we should design A3Ms as teaching aids and derive design guidelines for future model designers. We also expected to solicit feedback about existing A3Ms so we could improve the technologies to better support the use of A3Ms in classrooms.

We chose to achieve our study goals through design workshops. A workshop setting allows the researchers to engage with a relatively large number of participants in a short amount of time, allowing some one-on-one conversation as well as discussion among the participants.

5.2.2.1 Participants

We recruited participants for two workshops, which were conducted in two state conferences of the Association for Education and Rehabilitation of the Blind and Visually Impaired (AERBVI (2019)), an organization for professionals who provide services to people with VIs. The two workshops followed a

same protocol but were hosted in different locations with different participants.

Sixteen TVIs (3 males, 13 females) participated at the first workshop, whose reported years of experience ranged from 3 to 40 years (mean = 21, SD = 13). One identified as low vision, two identified as blind, and 13 identified as sighted. Nineteen TVIs (5 males, 14 females) joined the second workshop. Their reported years of experience ranged from 2 to 41 years (mean = 17, SD = 12). One identified as low vision, one identified as blind, and 17 identified as sighted.

5.2.2.2 Apparatus and Materials

We printed seven 3D models from (Thingiverse, 2019). As shown in Figure 5.8, these models were (1) a map of the U.S. (Thing: 61210), (2) a rocket (Thing: 100070), (3) an anatomical heart (Thing: 932606), (4) an elephant (Thing: 182136), (5) a house (Thing: 270223), (6) a skull (Thing: 622390), and (7) a DNA



Figure 5.8: The seven printed models we used in the workshop study.

(Thing: 1281735). We used Makerbot and Ultimaker printers to print these models with PLA filament. To allow participants to document modifications and annotations on the printed models, we prepared Play-Doh, stickers, sticky notes, pens, markers, and poster boards.

To allow participants to experience A3Ms with existing annotations, we created three A3Ms and ran Talkit on three Macbook laptops. The three A3Ms were: Cell, Globe, Map, models used in Section 4.2.

5.2.2.3 Procedure

Each workshop lasted 1.5 hours. We began each workshop with a short presentation introducing 3D printing technologies, A3Ms, and the design of Talkit. Then, participants split into groups (G1 to G4 in the first workshop, and G5 to G8 in the second workshop). Each group was asked to design an A3M.

At the start of their design process, each group chose one model from the seven models we printed. The participants chose models based on their interests, and on a first-come, first-serve basis. Their choices are shown in Table 5.7.

Then, each group modified the model using materials like Play-Doh, and used markers and stickers to create interactive elements on the model. They brainstormed the type and content for these elements, and how students would interact with them. During the design process, they also explored our existing

Group Number	1	2	3	4	5	6	7	8
Chosen Model	Elephant	Heart	Map	Skull	House	Heart	Elephant	Map

Table 5.7: Different groups and their chosen models in the workshop study

A3Ms using Talkit.

At the end of the workshops, each group delegated one or two representatives to briefly present their designs.

5.2.2.4 Analysis

We video recorded the presentations and took photos of the modified models. Two researchers developed themes from the recording transcripts using axial coding (Saldana, 2015). We refer to members of each group by their group number (G1 - G8).

5.2.3 Findings

5.2.3.1 Model Modification

Participants provided ways to improve and modify the designs of 3D printed models.

Size

Four groups shared concerns about the printed models being too small and explained that their size could impede tactile exploration and understanding of the concepts related to the referent object. G1 and G7, who both chose the elephant model, thought the small model would mislead students. G1 said it would be better if they could “have something accompanying that model to show scale.” For example, a 3D print of the foot of an elephant. Similarly, G4



Figure 5.9: The participants used stickers to indicate where they want to add annotations. The top of the elephant model has some printing imperfections.

also wanted to have an anatomically correct skull to show the correct size of the skull. G8 wanted a larger map model so students could feel each of the states.

Texture

Participants were unsatisfied with the limited printing quality we had and the solid printing materials we used. For example, the printing imperfections resulted in ridges on the top of the elephant model (see Figure 5.9) and “stringy” surface on the heart model. G6, who chose the heart model, said the model “should be lovely to touch” instead of being “like rough sandpaper.” They wanted to use rubbery materials to make the texture of the model closer to a heart. G1 thought that “different parts [should have] different texture.” They also suggested printing the model in a higher resolution to avoid imperfections. The limited printing quality is mostly because of the printers we used. We expect future printers will have a better printing quality. New printing materials (e.g., ABS) will also make texture more realistic.



Figure 5.10: The participants modified the skull model. They used Play-Doh to indicate different tactile patterns on four components.

Tactile Differentiation

G2, G4, and G8 wanted to add tactile patterns on the models to make different components more distinct from one another. G4 said “there wasn’t much tactile discrimination between them (different components on the skull model),” and they used Play-Doh to indicate different tactile patterns they would add onto the model, as shown in Figure 5.10. G2 and G8 had similar thoughts on the heart model and the map model.

Visual Cues

Four groups wanted to add visual cues to their models for low vision students. G7 thought the elephant model should be in the color of an elephant instead of being black, because most students with VIs have usable vision. Both G2 and G6 wanted to make a multi-color heart model. For example, G2 thought the arteries and veins should be color-coded to demonstrate blood flow. G4, who indicated that tactile patterns were useful to blind students, thought higher contrast would help low vision students better understand the model.

Removable Components

Two groups were interested in removable components, which would allow students to explore the relationships between different components. For example, G5, who chose the house model, wanted to add a removable roof, so students could explore the inside of the house when taking off the roof. Similarly, G4 wanted to create a removable brain component inside the skull model. However, in prior work, most of the printed models researchers designed do not have removable components (Gual et al., 2011, 2012, 2014; Kolitsky, 2014; Stangl et al., 2014, 2015; Grice et al., 2015; Brock et al., 2015; Holloway et al., 2018); only a few of work (e.g., Kim and Yeh (2015) explored the importance of removable components. Our findings suggest that future work should consider the importance of removable components when designing A3Ms.

5.2.3.2 Interactive Elements

After modifying the physical models, the participants designed interactive elements for their models. In addition to audio labels and detailed description that were widely deployed in prior work (Götzelmann, 2016; Reichinger et al., 2016; Taylor et al., 2016; Shi et al., 2016b, 2017b; Holloway et al., 2018), participants also used audio effects and animations.

Labels and Description

In general, the participants were all in favor of annotating model components with name labels, which could be followed by detailed descriptions when queried by users with VIs. They labeled each component of their models, and the number of labeled elements ranged from four (on the skull model by G4) to

12 (on the heart model by G6). They also added detailed descriptions about each component's function (e.g., the function of arteries), size (e.g., the size of an elephant leg), texture (e.g., the texture of elephant skin), related metrics (e.g., the population of a city), and surrounding components (e.g., surrounding states). G1 wanted to add a story to the elephant model in addition to factual descriptions.

Audio Effects

Six groups used audio effects in their interactive elements. Both G1 and G7 wanted their A3Ms to play elephant sounds when a student touched the trunk on the elephant model. G2 wanted the sound of the heartbeat. G6 associated heartbeat sounds with gestures, suggesting that "two taps (tapping on the model twice) would [play] like an active heartbeat, uh, maybe three taps would be like a very active heartbeat." For the map model, G3 added state songs and the sound of a state bird, and G8 wanted to have the sound of a state animal.

Animations

In addition to audio and visual feedback, G2 wanted an animation of a beating heart on the heart model.

5.2.3.3 Interaction Techniques

Participants suggested different input and output techniques for the interactive information on the models.

Input Techniques

Gestures were the most common input technique suggested. For example, performing single and multiple taps on a model (G1 - G6, G8) to get audio labels and detailed descriptions, respectively. G4 designed buttons near their model, with which a student could retrieve and record audio annotations. In addition, some participants wanted speech input. For example, G3 wanted simple commands like “Song” to activate audio effects on the map model. They also mentioned more conversational commands like “What’s this state’s song?” Overall, these input techniques are similar to what we found in the first section of this chapter, where we designed input techniques directly with adults with VIs.

Output Techniques

In addition to auditory feedback (e.g., annotations and audio effects) and visual feedback (e.g., animations) commonly mentioned by participants, G1, G4, and G6 also wanted to connect refreshable braille displays to the A3Ms as one potential output channel. The display could provide longer descriptions that students could access in braille. For example, G6 wanted to use a braille display to describe the complicated structure of the heart model.

5.2.3.4 Pedagogical Considerations

When describing how they would use A3Ms in their lessons, participants provided four suggestions.

Overview of a Model

The participants pointed out that students should have an overview of a model before going into more complicated details. G3 said that some students focused on details and missed the overview when studying a map model. G4 thought that students should know “the total number of things to find” on the skull model. The overview could be provided by a TVI or through the A3M.

Changeable Digital Content

The participants wanted to reuse a same printed model with different digital content. For example, G4 wanted to “erase that information (interactive elements) and use it with a different student.” They also suggested changing the content on a model as a student advances in their studies.

Connections between Models and Referent Objects

As discussed previously in Section 5.2.3.1, most 3D printed models had a limited size and a low-fidelity texture, which could lead to a misunderstanding of their referent objects. During the study, we used consumer-grade printers from Makerbot (2019) and Ultimaker (2019) to fabricate 3D models. These printers are popular in schools but have a limited printing quality. Thus, the models we prepared only have low-fidelity textures. The participants wanted their students to be aware of the difference between a model and its referent object and provide different solutions. For example, G4 wanted to bring a real skull along with the model. With the improvement of 3D printing technologies, the participants might have different opinions towards models and referent objects. However, the models we prepared in the study demonstrated the current situation that researchers should be aware of.

Student-Made Annotations

G4 wanted to enable students to record their own annotations. They said, “if he tapped the button or double tapped, he could add his research to that.” They thought that these annotations could be used in presentations or a science fair.

5.2.3.5 Feedback for the Talkit Application

Some participants tried the Talkit application and felt the setup of the desktop application was inconvenient. For example, the current camera position might lead to a high cognitive load for students with VIs. As shown in Figure 5.11a, the Talkit application used the webcam of a laptop to track a model and gestures. In this situation, as G3 said, “the kid has to reorient themselves when they put the map in front of them and because so many of our kids also have perceptual issues, that might be tough.” They suggested setting the camera above the user or right in front of them, facing the same direction as the student (Figure 5.11b).

A few participants discussed the disadvantages of the 3D tracker, which is a $2 \times 2 \times 2 \text{ cm}^3$ cube covered with five fiducial tags. The tracker enabled the Talkit application to locate the position of the model. However, the tracker could be bulky on some models.

5.2.4 Discussion

The goal of this study was to derive design guidelines for A3Ms from TVIs. Thirty-five TVIs participated in this study and provided suggestions about the designs of A3Ms from different aspects, such as model modification, interactive



Figure 5.11: (a) Participants tried existing A3Ms with Talkit application running on a Macbook. The application uses the webcam of the Macbook. (b) Unsatisfied with the orientation settings, a participant rotated the webcam to an egocentric way.

elements, interaction techniques, and pedagogical considerations. Although we successfully collected data from a wide range of TVIs during the workshops, the data was mostly about conceptual designs that were not implemented or used with students. Moreover, the TVIs were constrained by the seven 3D prints that we provided as a foundation. The fact that participants were not able to choose models from a larger set was one limitation in the first study. Besides, we only recorded the presentations for each group, and not their in-group discussions. Those in-group discussions could have provided deeper insights.

To address the limitations in the workshop study, we need to gather more comprehensive findings by deploying A3Ms with TVIs in their classrooms. Thus, we conducted another study, where we improved Talkit and worked with three TVIs to design and deploy new A3Ms. In the next section, we will describe the new software application and the study. Because these two studies are highly connected, we will discuss the design guidelines of A3Ms with the findings from these two studies together in the end of this chapter.

The workshop study also exposed the limitations of the Talkit application. First, Talkit does not support visual content. As described in 5.2.3, visual cues and animations can help low vision students learn concepts better. Second, the Talkit application only spoke descriptive texts, although audio effects are also helpful to students with VIs. Third, the setup of the application should be easier. In Section 5.2.3.5, the TVIs from the workshop study pointed out that the current setup would lead to high cognitive load for students. Last, the application should support other forms of trackers (e.g., 2D fiducial tags). As mentioned in Section 5.2.3.5, 3D trackers are bulky and may not be necessary for some models. We should provide 2D trackers as an option as well. These limitations drove us to develop a new application, Talkit++, which will be described in Section 5.3.2.

5.3 The Co-Design Study

5.3.1 Introduction

Although the workshop study in Section 5.2 helped us gather rich feedback from a wide range of participants, it was limited to conceptual designs that were not implemented or used with students, and the TVIs were constrained by the models that we selected.

To address the limitations of the workshop study, we conducted a co-design study, where we collaborated with three TVIs individually to design and deploy sample A3Ms over seven weeks. First, we improved the prior system and developed Talkit++, a mobile application that supports the use of A3Ms in an educational setting. Then, we worked with the TVIs to design three A3Ms they chose: the Plane model, the Volcano model, and the Map model. These A3Ms, which can be deployed using the mobile application, take gesture and speech commands as input, and support auditory (e.g., labels, detailed descriptions) and visual feedback (e.g., animations, color highlights). At the end of the design process, the TVIs used and evaluated the A3Ms with their students. The students learned how to use A3Ms quickly and enjoyed the auditory and visual feedback. They understood the representational concepts better and demonstrated increased independence.

This section will start with the design and implementation of Talkit++. Then, we will describe the design and results of the co-design study.

5.3.2 Talkit++

To conduct the co-design study and deploy A3Ms in classrooms, we developed a mobile application, Talkit++, as shown in Figure 5.12. When designing the app, we considered the proposed design described in Section 5.1.5, and incorporated TVIs' feedback described in Section 5.2.3. These studies exposed the limitations of Talkit and inspired the new features in Talkit++.

Talkit has several limitations. First, Talkit does not support visual content. This is informed by the findings from Section 5.2.3. As described in page 113 and page 115, visual cues and animations can help low vision students learn concepts better.

Second, the Talkit application only spoke descriptive texts, although audio



Figure 5.12: A visually impaired student is using a mobile application to learn the interactive Plane model. The application speaks audio information when the student explores the model and highlights different components on the model with colors for low vision students. The iPad is on a stand so that the camera captures the student's interactions with the model.

effects are also helpful to students with VIs. As described in page 115, the TVIs from the workshop study showed interests in audio effects (e.g., the sound of a heartbeat). These audio effects can introduce a wide range of information to students with VIs.

Third, the setup of the application should be easier. In Section 5.2.3.5, the TVIs from the workshop study pointed out that the current setup would lead to high cognitive load for students. The orientation of the cameras may confuse students. Thus, we need to find an easier way to configure the application. We think converting Talkit into a mobile application will reduce the current obstacles. Mobile applications can be easily installed and the orientation of a smartphone camera can be adjusted.

Last but not least, the application should support other forms of trackers (e.g., 2D fiducial tags). As mentioned in Section 5.2.3.5, 3D trackers are bulky and may not be necessary for some models. We should provide 2D trackers as an option as well.

To address these limitations, we designed new features for Talkit++. The previous version, Talkit, tracks finger gestures using a red sticker, locates a model using a physical tracker, and provides audio labels and detailed description. The Talkit application also recognizes simple speech commands like “more.” Talkit++ inherits these features and has several new features:

- *Visual Cues*: The application visually highlights the components on a model (see Figure 5.12).
- *Animations*: Users can trigger animations using a button on the touch-screen interface.

- *Audio Effects*: Designers can add audio effects on A3Ms.
- *2D Trackers*: In addition to 3D trackers, Talkit++ also supports tracking models using 2D trackers, which are less intrusive.
- *iOS Compatibility*: The application runs on iOS devices. The user interface of the application is shown in Figure 5.13.

Talkit++ was developed in Swift and C++. We cross compiled computer vision libraries like Chilitags (Bonnard et al., 2013) and OpenCV (Bradski and Kaehler, 2008), and called them in Swift. Talkit++ uses SpeechRecognizer (Apple, 2019d) to detect speech commands. As for output, Talkit++ uses a text-to-speech engine (Apple, 2019b), plays audio effects through AudioPlayer (Apple, 2019a), and displays animations using UIImage (Apple, 2019e).

To use an A3M, a user with VIs must put a red sticker on his fingernail and run Talkit++ on an iOS device (e.g., an iPhone or an iPad). As mentioned in Section 5.2.3.5, a fixed camera position may lead to a high cognitive load as a

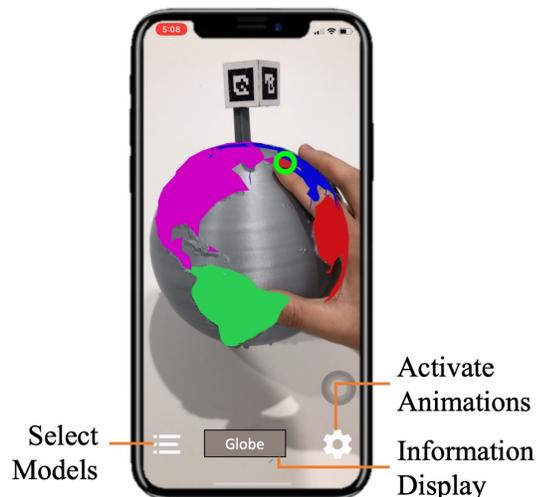
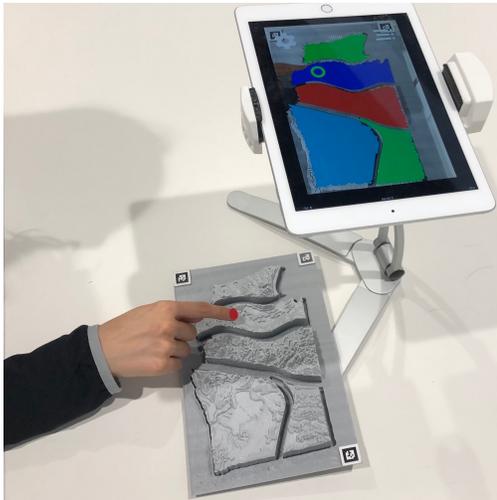
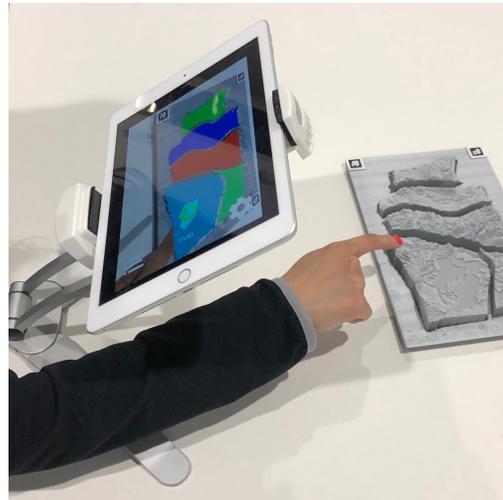


Figure 5.13: The user interface of the Talkit++ application, which is accessible with the built-in VoiceOver screen reader.



(a) A user positions the iPad above the model. In this way, she can explore the model freely, without any object between herself and the model.



(b) A user positions the iPad between herself and the Map model so that she could see the highlighted interactive elements while exploring the model.

Figure 5.14: Two ways a user could position an iPad using a stand.

student explores an A3M. When use Talkit++, the user can put the device on a stand to adjust the camera position of the device (see Figure 5.14).

In the current system, the student needs to select the digital content that works with the model using a button on the application. The current system cannot detect which model the user is using. Thus, the user has to provide this information. In the future, users could select models using speech input. In addition, the system can also detect a model automatically using the model's trackers and shape. Once selected, the model speaks the overview introduction. Then, he can explore the printed model tactilely. When his finger touches an interactive element, the model plays audio labels and audio effects. He can also say "more" to get a more detailed description.

5.3.3 Study Design

The goal of the co-design study was to explore the design of A3Ms in practice. While the workshop study aggregated suggestions from conceptual designs, this study allowed us to work with individual TVIs in-depth and deploy A3Ms in their classrooms. Both studies answered our research question: how should we design A3Ms as effective teaching aids? In this study, we specifically wanted to explore (1) How would TVIs design and use A3Ms in their lessons? (2) Are the A3Ms designed by TVIs useful and effective in the classrooms?

We conducted the co-design study with an instructional design process. Instructional design is the practice of developing learning experiences (Merrill et al., 1996). The development of technology has shaped the theories and guidelines of ID over time (Reiser, 2001). In our case, instructional design has two components: the design of A3Ms and the design of teaching plans. We worked with TVIs to develop and test these two components over seven weeks. Through this process, we designed three A3Ms along with teaching plans, and gained feedback from TVIs and their students with VIs.

5.3.3.1 Participants

Three TVI participants, denoted as T1 - T3, joined our study. All participants were female, and their ages were 52 (T1), 34 (T2), and 24 (T3). Their years of experiences as TVIs were four (T1), one (T2), and one (T3). We recruited them through a Master's level personnel training program for TVIs. The participants were actively serving with VI students while completing their degree program to get certified.

In total, ten students participated. Instead of reporting the information of each student, we report the aggregated demographic data of the students to preserve their privacy:

- T1 planned to work with six students, whose ages were: eight, nine, nine, nine, 10, and 11. Four of them were blind, and the other two were low vision. Three of them had mental disabilities.
- T2 planned to work with three students, whose ages were: 16, 18, and 19. Two of them were blind, and the other one is low vision. All of them had mental disabilities.
- T3 planned to work with one blind student, whose age was 11. The student did not have other disabilities.

5.3.3.2 Procedure

The entire study lasted seven weeks, and we followed ADDIE (Piskurich, 2015), a commonly used process in instructional design, to develop A3Ms. ADDIE is an acronym for five phases: Analysis, Design, Development, Implementation, and Evaluation. A researcher, who was not co-located with the participants, scheduled weekly 30-minute remote meetings with each TVI. In some cases (e.g., design iteration), they had extra short remote meetings. Besides the meetings, they sent emails and short messages. The researcher spent around 12 hours in total meeting with the three TVIs during the study.

Week 1: Analysis

The goal was to (1) introduce the research project to the TVIs and (2) analyze and articulate the teaching needs. The researcher introduced A3Ms and demon-

strated the capabilities and limitations of A3Ms. Then, the TVIs were asked to think about who they would teach and potential concepts they wanted to teach with A3Ms. After the meeting, the TVIs were instructed to search online and come up with three potential teaching concepts.

Weeks 2-3: Design

The goal was to (1) design a physical model for each TVI, and (2) add interactive elements for the model. First, the researcher worked with each TVI to choose one teaching concept and design a corresponding model. They chose a concept based on design complexity, printing feasibility, and feedback from other peers. Then, the researcher helped the TVIs design their selected models. The TVIs used similar digital models, drawings, and diagrams to illustrate their ideas.

After designing the physical models, the TVIs designed interactive elements. They were asked to provide the textual content of the overview introduction, audio labels, and detailed descriptions. Optionally, they also added audio effects and animations. They provided multimedia materials (e.g., audio files, YouTube videos) to the researcher if they wanted these features incorporated in their A3Ms.

Week 4: Development

The goal was to develop A3Ms and teaching plans. The researcher modified models found online to meet the TVIs' designs and printed them out. When the researcher had design or printing issues, he discussed these problems with the TVIs and together they adjusted the models.

Then, the researcher added textual content, audio files, and animations to

the models. The audio files and animations were created from the materials that participants provided. The TVIs did not use 3D modeling software themselves during the study; the researcher iterated on the designs with feedback from the TVIs multiple times during the study.

During this phase, the TVIs were asked to develop their teaching plans for using A3Ms in their lessons.

Weeks 5-7: Implementation and Evaluation

The researcher and collaborating personnel university program provided research materials to the TVI participants. For each TVI, we prepared an iPad with the Talkit++ application, an iPad stand, finger stickers, trackers, and the selected printed model. An accessibility specialist and other researchers tested the usability of Talkit++ application before delivering it to the TVIs. The TVIs had at least two weeks to get accustomed to the application before handing it to their students.

The TVIs implemented their teaching plans with their students. After each teaching session, the researcher interviewed TVIs. In the last lesson with A3Ms, the TVIs asked their students to provide feedback on the A3Ms.

5.3.3.3 Data Collection

We recorded all the remote meetings, and the TVIs provided their students' feedback on A3Ms. We only used the data provided by the TVIs because of feasibility and privacy concerns. Two researchers transcribed and coded the recordings we collected from TVIs.

5.3.4 Design Deliverables

5.3.4.1 Initial Concepts and Model Selection

The participants chose models according to their teaching curricula and students' individual capabilities. Initially, each of them proposed three models and narrowed down their choices based on feedback from their TVI colleagues, braillists, and the researchers.

T1 wanted to teach the structure of the earth, the structure of a flower, or the structure of a volcano. The structure of a flower was too fragile to print. After conferring with her classmates, she decided to design an interactive volcano model, which would fit her teaching agenda and be more suitable for her students.

T2 wanted to teach the concept of an airplane, or animals like birds and elephants. After discussing with her peers and students, she chose an airplane model.

T3 initially proposed a model for conducting electricity experiments. She wanted to let students assemble printed components (e.g., a printed battery and a printed bulb) together and then get interactive feedback. This idea required interactions between 3D prints, which was not well supported by the current Talkit++ application. Besides the experiments, T3 was also interested in teaching with an airplane model or a printed map. We chose a map model to avoid duplicates.

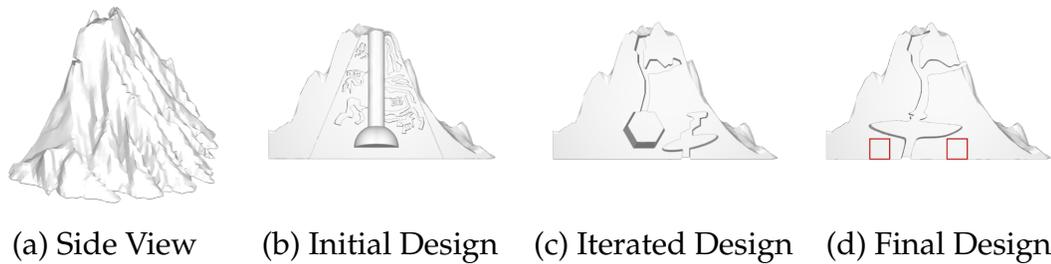


Figure 5.15: Different designs of the Volcano model. The final design includes two 2D trackers, which are highlighted in red.

5.3.4.2 Model Iteration

The Volcano Model

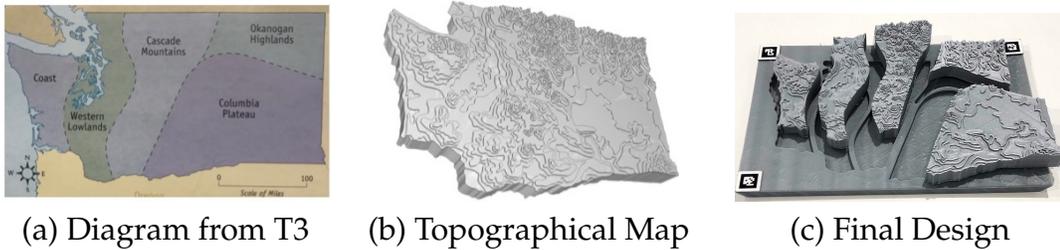
T1 wanted external tactile textures of a mountain with the internal structure of a volcano. The researcher proposed an initial design by combining Thing:1316498 (mfritz, 2016) and Thing:1370593 (kristinaprofitt1, 2016), as shown in Figure 5.15. However, T1 thought the model was too complicated. Thus, we iterated on the design and replaced the internal structure with the design from Thing:1290606 (WaveSkyLord, 2016). In the final design, we made the model simpler. Since all the components are on one 2D surface, we used 2D trackers. We added two in case one of them was blocked by a students' hand. The dimensions of the model are $22 \times 15 \times 13 \text{ cm}^3$.

The Plane Model

We modified Thing:182252 (YahooJAPAN, 2013), as shown in Figure 5.16. The dimensions of the model are $23 \times 21 \times 6 \text{ cm}^3$. The application could still locate the 3D tracker when students rotate the model. T2 was really excited about the model and said, "This is something our students really could not see or experience in any other way."



Figure 5.16: The Plane model designed for T2 and her students.



(a) Diagram from T3

(b) Topographical Map

(c) Final Design

Figure 5.17: The materials and final design of the Map model.

The Map Model

T3 provided a diagram of a Washington State regional map, as shown in Figure 5.17a, to the researcher for reference. The researcher found a model with similar content from GrabCAD (2019). T3 wanted separate components for each region. The researcher iterated on the design of the model with T3 and designed a map puzzle. As shown in Figure 5.17c, the puzzle consists of five pieces and a base, which has three 2D trackers on its corner. A student could take the pieces off the base. The dimensions of the model are $26 \times 17 \times 4 \text{ cm}^3$.

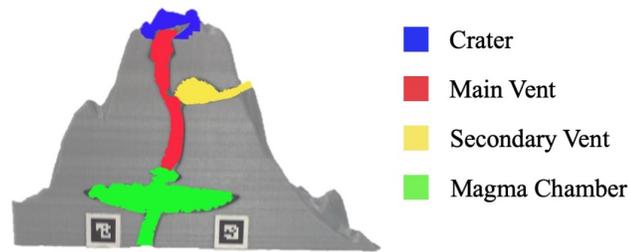


Figure 5.18: The interactive elements on the Volcano model.

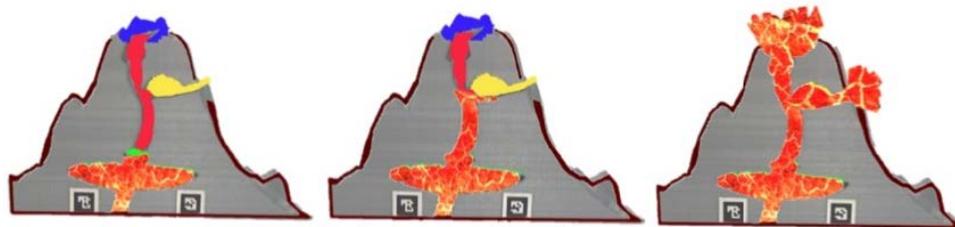


Figure 5.19: The animation of the Volcano model.

5.3.4.3 Interactive Elements

The Volcano Model

T1 designed audio labels and detailed descriptions for four components, as shown in Figure 5.18. In addition, she wanted an animation of a volcano eruption. The researcher designed an animation based on a YouTube video she provided, as shown in Figure 5.19. The animation also came with an audio introduction and could be triggered by touching a button on the iPad screen.

The Plane Model

T2 designed audio labels and detailed descriptions for seven components, as shown in Figure 5.20. She also designed three audio effects: (1) a recording of a pilot speaking for the Cockpit component, (2) the sound of people chatting for

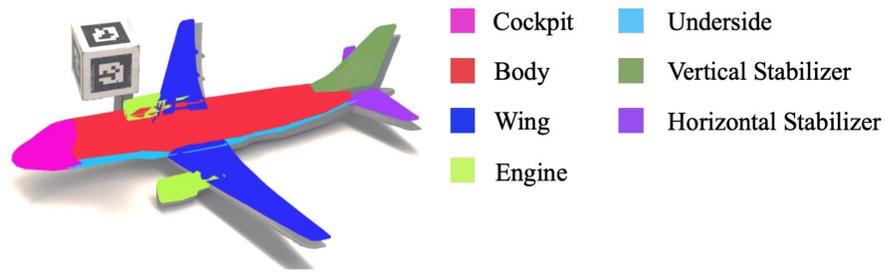


Figure 5.20: The interactive elements on the Plane model.

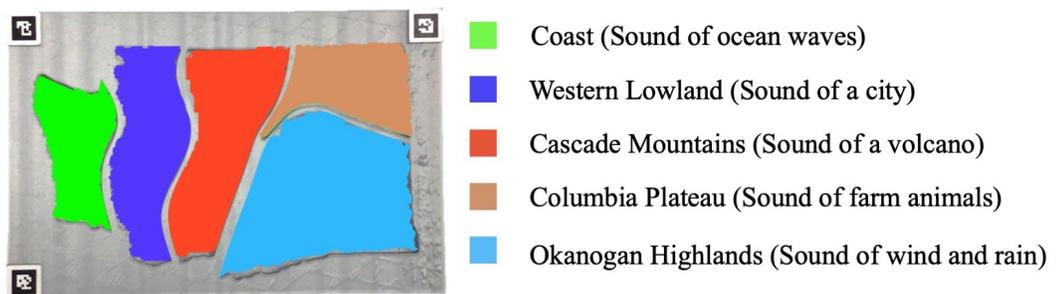


Figure 5.21: The interactive elements on the Map model.

the Body component, and (3) the sound of an engine propeller roaring for the Jet Engine component. The audio effects played after the audio labels, and the detailed descriptions could be retrieved by saying “more.”

The Map Model

T3 designed audio labels, detailed description, and audio effects, see Figure 5.21.

5.3.4.4 Teaching Plans

The Volcano Model

T1 planned to have two group sessions with her students. In the first session, she planned to introduce basic knowledge about volcanoes (e.g., what a volcano is and how it works). Then, she wanted to let her students explore the physical model and use the A3M to learn the detailed structure of a volcano. In the second session, she developed assessment prompts to have the students recall different components of the model and assess learning using A3Ms. In the end, she planned to evaluate the A3M with her students.

The Plane Model

T2 planned to have two group sessions with her students. Her teaching plan followed a similar structure to T1's plan.

The Map Model

T3 wanted to have three sessions with her students. In the first session, she wanted to introduce the basic knowledge of Washington State and its five regions. Then, she planned to have the students experience the physical model without any interactive features. In the second session, she planned to help her student set up and use the mobile application. In the third session, she wanted to let the student set up the application independently. She also wanted to assess her students and evaluate the A3M, as similar to T1's plan.

5.3.5 Findings

We aimed to understand how would TVIs and their students think about the use of A3Ms in classrooms. In general, all the TVIs thought the sessions were successful, and their students were excited about the A3Ms. T3 said “she (her student) was really, really excited about it and she thought it was really cool.” We further asked the TVIs to define “success” in terms of system performance, the design of interactive elements, the progress of their students, and students’ feedback.

5.3.5.1 System Performance

In general, the Talkit++ application performed well in the study. T3 said, “the actual technology, it was recognizing very quickly what she was touching.” However, some students encountered usability issues during the study.

Four out of six students working with T1 had difficulty activating interactive elements. Two students were not sure that “the sticker had to face the iPad,” and stick their fingers in the holes. The application couldn’t recognize their fingers without a visible red sticker. T1 thought it might be helpful to make a sticker that “goes all the way around their fingers.” Another student, who wore nail polish, had to fold her other fingers because the system misclassified the fingernail polish as a sticker. One student moved his finger all over the place quickly, and the application could not detect his moving fingers. Reflecting on the all sessions, T1 said most students managed to use to the system in the end.

T2 had a “very dependent” student who needed a lot of guidance to find

the different parts of the model. She ended up providing hand-over-hand assistance.

T3's student kept moving the model away from the application's field of view. She said that "the student kind of gradually would shift the model around... The iPad would maybe stop recognizing." T3 ended up using her laptop to set boundaries for the student.

5.3.5.2 Interactive Elements

The TVIs thought their students enjoyed both the auditory and visual elements but wanted more content.

All the TVIs liked the implemented auditory feedback. T2 thought the audio effects were unique and said "the sound of the people talking and the motors and the pilots inside the cockpit. Those are some elements that we couldn't produce otherwise [without digital devices]." T1 thought it would be better if the A3M had more information, because she thought her students "got bored kind of fast." She suggested additional layers of information and more detailed explanations.

T1, the only TVI that had an animation on her model, thought the animation was attractive and effective. She taught the A3M in a group session, and she asked her students to try the model in turns. She found one low vision student was attracted to the animation, and "kept sticking her head in the way when the first girl was using it." Even for students who are totally blind, the animation was still useful, because "it's still interesting to listen to [the audio of the animation]."

5.3.5.3 Learning with A3Ms

The students became familiar with A3Ms quickly and used them independently. For example, T1 said her students learned how to use the models in her first session quickly and said, “They didn’t need more than a few seconds of instruction to do it.” In her second session, she let her students explore the model independently, and “they all remembered how to use it.” Some students were even able to set up the A3Ms by themselves. T3, who tried to let her student launch Talkit++ and put a sticker on by herself, was surprised that the student finished the setup quickly. She said excitedly, “Wow, a fourth grader could use this independently after a couple of times practicing with a teacher.” She attributed this to the accessible design of the application, which allowed the student to use VoiceOver, the iOS built-in accessibility screen reader. T3 thought, with practice, the student should be able to setup the stand for an iPad and position the model independently.

A3Ms afforded a unique, advanced learning experience with tactile, auditory, and visual feedback. For example, T1 said, “for my kids who can’t see at all, they wouldn’t be able to understand a diagram of the inside of a volcano. So it helped them to be able to feel what the outside is like, and the inside parts.” T2 liked the design of the Plane model and said, “Neither of them had been inside an airplane before using the 3D model, it was a good way to give them a concept of the shape of an airplane and how it works.” All three TVIs reported that their students understood the taught concept better after using the A3Ms. T3 said her student knew all the names and descriptions of the five regions on the Map model in their final assessment.

5.3.5.4 Students' Feedback

The TVIs asked the students about their feedback on A3Ms.

All students liked A3Ms and thought the model helped them understand the taught concepts. They felt the models were better than traditional tactile graphics and wanted to use A3Ms again in the future.

When asked what other concepts they wanted to learn using A3Ms, the students showed a wide range of interests. The student working with T3, aged 11, was interested in a "Tactile Town," where the application could name shops like Starbucks. The students who learned the Plane model were interested in cars and motorcycles. Among the students working with T1, one was interested in learning "how rain comes down," while the others were interested in planets, the galaxy, a space shuttle, and a model of Hawaii.

The students were asked to compare using A3Ms with teacher instruction only. Four students explicitly said they would prefer learning with A3Ms independently. T3's student said, with the help of A3Ms, she wouldn't have to take someone's time and become more independent. which was echoed by one of T1's students. The other two students thought the auditory and visual feedback, which were designed by TVIs, were more intriguing compared to a TVI.

5.3.6 Discussion

A3Ms provide an interactive learning experience. With A3Ms, students could gain a lot of knowledge from auditory and visual information, which is not

available through traditional teaching aids (e.g., tactile graphics). In prior studies, A3Ms were also proven to be capable of facilitating space and text memorization (Giraud et al., 2017) and providing an overview understanding of concepts (Holloway et al., 2018).

We think A3Ms could lower the teaching load of TVIs, and TVIs should be highly involved into the design and implementation process of A3Ms to provide customization suggestions and domain knowledge. A3Ms changed the way the TVIs interacted with their students. For example, T3 deliberately skipped some content in her lesson, allowing her student to learn this content by interacting with the model. All TVIs reported they had more time to observe their students as they interacted with the models. Based on these observations, they adjusted their teaching plans on the spot.

The participation of TVIs is important in the design process of A3Ms because they understand the diverse needs of their students. In our study, the TVIs customized the content of A3Ms to their students and the students' disabilities. In practice, students with VIs usually have other disabilities (Batshaw, 2002). While we could study how to design A3Ms for students with different disabilities in long term, the adoption of these models will still highly rely on the help of TVIs.

TVIs could help students benefit from the advantages of A3Ms and avoid potential challenges using their teaching techniques. For example, in the workshop study (Section 5.2.3 on page 111), the participants were dissatisfied with the limited sizes of the printed models. However, in the co-design study, the TVIs helped their students understand the scale of the model in relation to the referent object, and there was no feedback regarding model sizes.

Based on the feedback from the co-design study, Talkit++ could be further improved. First, the application could be more responsive by running GPU-accelerated algorithms. Second, we could deploy a skin color-based tracking method to avoid the need for stickers. Third, we should provide better audio cues to prevent students from moving models away. Also, the speech input could be more conversational, allowing students to ask the model questions about what it represents and its content.

5.4 Summary and Design Guidelines

In this chapter, we present three studies that explore the interaction design of A3Ms. These three studies are: the elicitation study, the workshop study, and the co-design study. The first two studies also contributed to the design of Talkit++, an iOS application that enables A3Ms.

In the elicitation study, we recruited 12 legally blind participants to understand people with VIs' exploration behavior with 3D printed models and elicit interactions for future A3Ms. In addition, we proposed a design of A3Ms. In the workshop study, we led two design workshops with 35 TVIs, who modified sample models and added interactive elements to them. In the co-design study, we worked with three TVIs to design three A3Ms in an iterative instructional design process, and the TVIs used the A3Ms to teach their students at the end of this process.

These studies explored the design of A3Ms from different perspectives. The elicitation study engaged end-users, people with VIs, to explore the design space of interaction techniques. The study results showed a series of input techniques for future A3Ms. The workshop study and the co-design study focused more on educational settings. The workshop study involved more participants but only explored the designs conceptually, while the co-design study explored the research question through an in-depth design process with fewer participants. All studies shed light on the design of A3Ms.

Based on the findings from the three studies, we derived the following design guidelines. While prior work in A3Ms provided design implications, we

are the first who organize them into systematic design guidelines. Compared to prior work, these guidelines provide practical and useful suggestions for A3Ms. These aspirational guidelines will enable future model designers (e.g., TVIs, O&M specialists) to create better A3Ms, and inform investigators about future research initiatives in digital fabrication and accessible educational technology.

A3Ms should have effective tactile features

- Use different printing materials to make the texture of a model similar to its referent object.
- Adjust the printing settings (e.g., speed, temperature, resolution) to avoid printing imperfections that may be confused for tactile textures or model components.
- Use tactile patterns to make the different components of a model more distinct from one another.
- Avoid adding redundant tactile features to a model, keep tactile information simple, clear, and salient.
- Use removable components to allow users to explore the relationships between a model's different components.

A3Ms should contain both visual and auditory content

- Use multi-level information to indicate the name of an element and its detailed description.
- Add audio effects and animations to provide additional information for students.
- For low vision students, use visual cues that direct their attention to salient parts of a model and high contrast colors to help them distinguish between

model components.

A3Ms should consider pedagogical methods

- Use multi-level information to indicate the name of an element and its detailed description.
- Provide an overview of an A3M to students before going into details.
- Change the content of a model to meet the learning progress of students.

CHAPTER 6

DESIGN TOOLS

In Chapter 4, we discussed different sensing technologies that enable A3Ms. In addition, Chapter 5 explored the design space of A3Ms. In this chapter, we will focus on design tools that allow non-expert designers to create A3Ms.

Designing A3Ms is a challenging task for education practitioners like TVIs. Previously, to create an A3Ms, designers usually need to design a model from a sketch, assemble electronics and wires, and sometimes hard-code interactive elements in software. These tasks are time-consuming and require a lot of modeling and fabrication skills. However, most TVIs and O&M specialists are not trained as professional designers, and they do not have the required skills to create A3Ms.

To address these barriers, we aim to design easy-to-use tools that enables non-expert designers to add annotations to 3D models. First, we designed Markit, a Blender add-on. It provides step-by-step guidance for creating A3Ms for non-experts. After importing a downloaded model into Markit, a maker can add a tracker scaffold and associate textual annotations with elements on the model. Markit has a graphical user interface (GUI), where users perform all interactions using a mouse and keyboard shortcuts. Designing A3Ms with Markit are visually-demanding tasks with its GUI. However, a lot of TVIs are visually impaired and they would have to be dependent on sighted people to use Markit.

To make the design process of A3Ms more accessible, we created Molder. Molder supports the design of printed tactile maps in particular, but it also rep-

resents a paradigm for designing other printed tactile materials as well. To design an augmented map model, a designer creates a draft map model using Molder and prints the model. Then, she uses Molder to modify the map model by directly manipulating the printed draft model. Molder provides auditory feedback and high-contrast visual feedback to assist the designer in the design process.

In this chapter, we will describe the technical details and evaluation studies of these two design tools.

6.1 Markit: A GUI-Based Design Tool

6.1.1 Introduction

Markit is a tool for creating augmented models from online resources. With the help of the maker movement, there are a lot of free online models available. Markit enables designers to convert these free models into A3Ms. To create an A3M, a designer imports a free online model into Markit and adds a tracker scaffold, as shown in Figure 6.1. Then, she selects different areas on the model and associates them with interactive elements.

These augmented models can be used in the interactive applications presented in Section 4.2 and Section 5.3.2. A user with VI can print the augmented model and slide a tracker on the printed scaffold. Then, the user launches the Talkit or Talkit++ application. He accesses the associated information by touching the model and following the application's verbal cues.

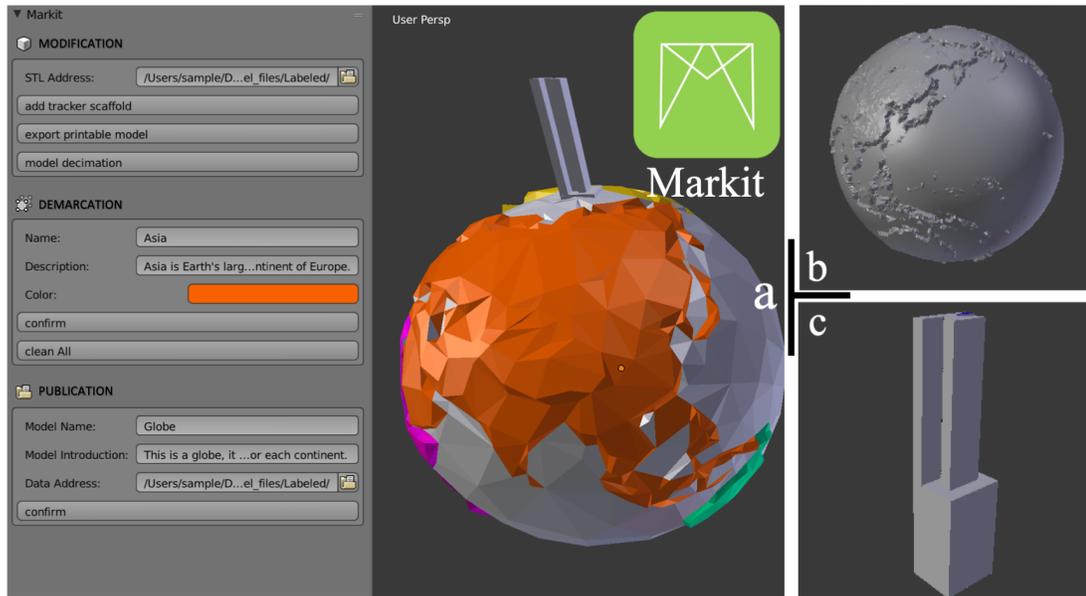


Figure 6.1: A designer downloads a model of a globe (b) and imports it into Markit. She adds a tracker scaffold (c) and augments areas on the model with textual information (a).

In this section, we will start with the design and implementation of Markit. Then, we will describe a user study that evaluates the usability of Markit.

6.1.2 The Design and Implementation of Markit

6.1.2.1 Interaction Flow

Markit provides step-by-step guidance for creating augmented models for non-experts. As shown in Figure 6.2, a user follows three steps to label an existing model, which can be downloaded from online resources:

- *Modification.* The user adds a tracker scaffold on the model. Markit exports

the printable STL file of the modified model and prepares a decimated model for future edits.

- *Demarcation*. The user marks an element with a specific color and annotate it with a name and detailed description.
- *Publication*. The user uses Markit to export the model's annotation data, which can be read by Talkit and Talkit++.

6.1.2.2 Implementation

Markit is an add-on for Blender (2019), which includes basic modeling functions like importing and exporting models, face selection, and 3D navigation. Most Blender's built-in functions have corresponding Python APIs, which allowed us to edit 3D models using Python scripts.

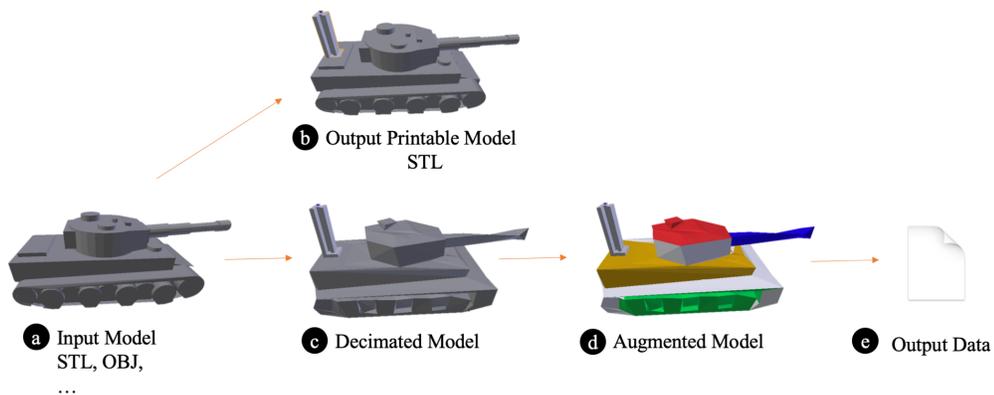


Figure 6.2: For an input model (a), a user uses Markit to add a tracker scaffold. Markit outputs a printable model in STL format (b) and decimates the model (c) for future edits. Then, the user marks the model in colors and adds annotations on it (d). Finally, Markit publishes the model's annotation data (e).

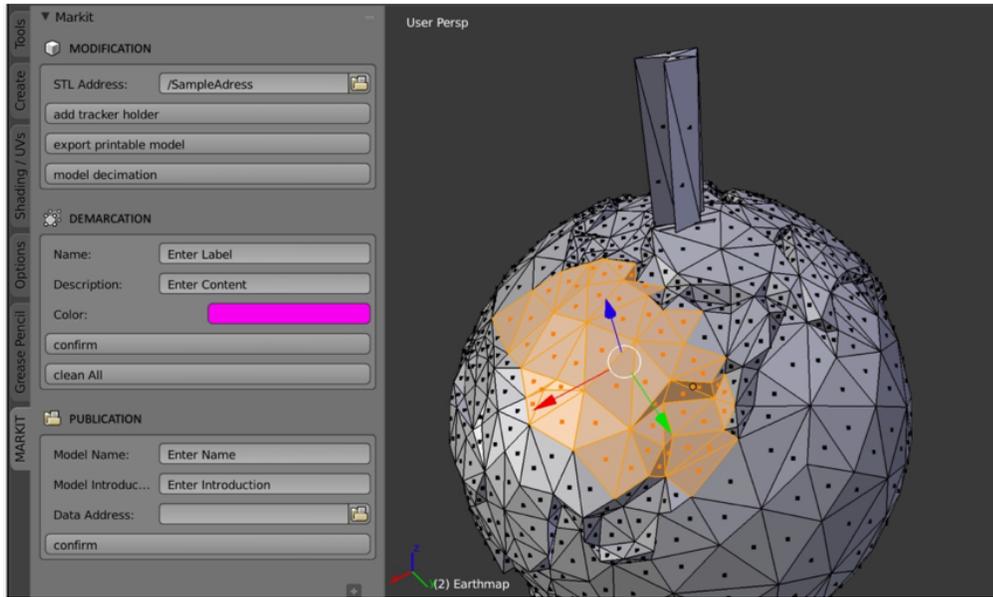


Figure 6.3: The Markit user interface. On the left is the Markit toolbar, and on the right is a perspective view of a working model. A user has selected several faces, which are highlighted by Blender in yellow, and will mark them with the information in the Demarcation section.

6.1.2.3 User Interface and Functions

As shown in Figure 6.3, Markit has three sections in its toolbar on the user interface: Modification, Demarcation, and Publication. A user works on an imported model by following these sections one by one. These sections represent the three steps described in Section 6.1.2.1: (1) modifying the design of a model, (2) marks the interactive elements on the model, and (3) publishing model data.

For an imported model, a user uses the functions in the Modification section to add a tracker scaffold, export printable model, and decimate the model. Markit loads a tracker scaffold when the user presses the first button in this section. Then, the user drags the holder to an appropriate position on the model and presses the export button after entering the destination path of the export

file. Since some complicated models with millions of triangular faces can slow the processing speed of Talkit and Markit, we use a decimation function to reduce the number of faces and speed up the toolkit. An expert user can use Blender’s built-in decimation filters to get customized decimation results.

After the Modification section, the user marks the triangular faces of the model as hotspots in different colors and adds annotations. In the Demarcation section, the user selects several faces of an element, and enters the name, the description, and the color of the element. The Markit dyes the faces with the chosen color. If the user makes a mistake when she marks faces, she can undo the previous actions or clear all marking data.

In the Publication section, the user enters the name and the introduction of the model, as well as the export path of the annotation data. Markit searches for all faces, associates them with particular annotations, and organizes them into a single file that is readable by Talkit and Talkit++. In the prototype, the Publication section exports a Python pickle file. For Talkit++, designers can put the links of audio files or video files to the description textbox.

6.1.3 Evaluation

The target users of Markit are education practitioners and hobbyists who enjoy fabricating objects to solve problems that are often socially-motivated. As such, we recruited people who considered themselves amateur “makers” and had some experience with digital fabrication. The goal of this study is to test the usability of Markit.

6.1.3.1 Method

Participants

Eight sighted volunteers (4 male, 4 female) joined our study, denoted as P1-P8. Seven participants had at least college degrees, and one participant was pursuing her bachelor's degree. Five participants (P2-P6) had experience with modeling software but only one (P6) used modeling software in the last year, and four of them had experience with 3D printers except P3.

Apparatus and Materials

We ran Markit on a 13-inch Macbook. Participants used the Macbook's trackpad and display to interact with Markit. We instrumented Markit and logged participant interactions. We also created an instructional pamphlet to facilitate the training session. The participants could refer to the pamphlet during the study. We provided participants with a PowerPoint presentation that indicated which elements should be annotated and with what information.

Procedure

During the study, we trained each participant how to use Markit and instructed them to augment models using Markit. We used four models in the study, as shown in Figure 6.4.

The study began with a training session. We introduced Markit and used the Cell model as an example to teach a participant how to augment a model. At the beginning of the study, we showed participants how to navigate the 3D views of Blender. Then, we taught them how to drag and move a tracker scaffold to add

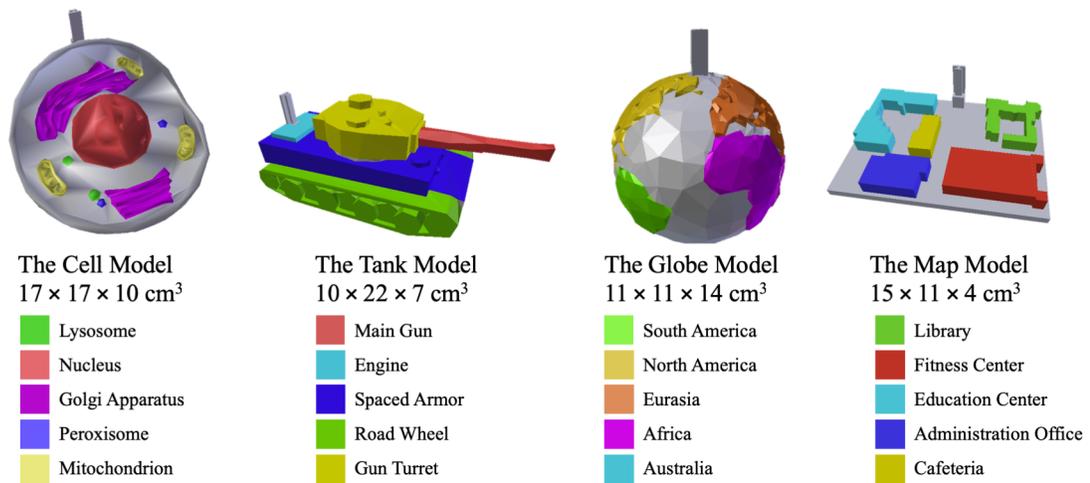


Figure 6.4: Four models used in the evaluation study of Markit. The four models are same as the ones in Figure 4.16. The models and their elements are highlighted in color, and their names and dimensions are listed in the figure as well.

it to a model. In the marking section, the participant learned how to select and deselect faces, and mark faces in the Cell model. The participant learned how and where to put the required information in Markit. At the end of the training session, the participant was asked to mark “Lysosome” and “Nucleus” on the Cell model to consolidate the skills she learned. Depending on the participant’s skills, the training session took about 10 to 20 minutes.

After the training session, we asked each participant to mark two elements on the Globe, Tank, and Map models. We presented a participant with one model at a time and explained the model along with its two assigned elements. For each model, we asked the participant to mark the first two elements listed in Figure 6.4, and the participant needed to complete all three sections in Markit by herself. We counterbalanced the order of the three models across participants with a Latin Triangle. This session took less than 30 minutes.

At the end of this study, we conducted a short interview with a modified Standard Usability Scale (SUS) (Brooke, 1996) to collect subjective feedback.

6.1.4 Results

6.1.4.1 The Time Needed to Add Annotations

Across all models that participants made, the mean time spent to augment a model with two annotations was 275 seconds (SD = 70). This shows that a non-expert user can use Markit to augment a model within minutes. Figure 6.5 shows the mean times participants spent on different models. The Map model took less time than the others on average. P4 spent only 158 seconds, the minimum value, to annotate the Map model. We noticed that participants made fewer navigation operations when dealing with the Map model, which was essentially a 2.5D model. The Globe model took the longest time on average to annotated, with P7 spending 420 seconds (the maximum value) on this model. When labeling the Globe model, some participants were unsure about the boundaries of continents and kept checking their results, which led to longer times.

To understand how much time was spent on different sub-tasks, we broke down the total time by the different sections in Markit. Among the three sections, the time spent on the Publication section was more consistent across participants, mostly because it only involves text boxes and buttons. While in the Modification and Demarcation sections, participants needed to drag objects, rotate models, and select faces. These tasks require more spatial navigation skills with Blender's selection and navigation operations.

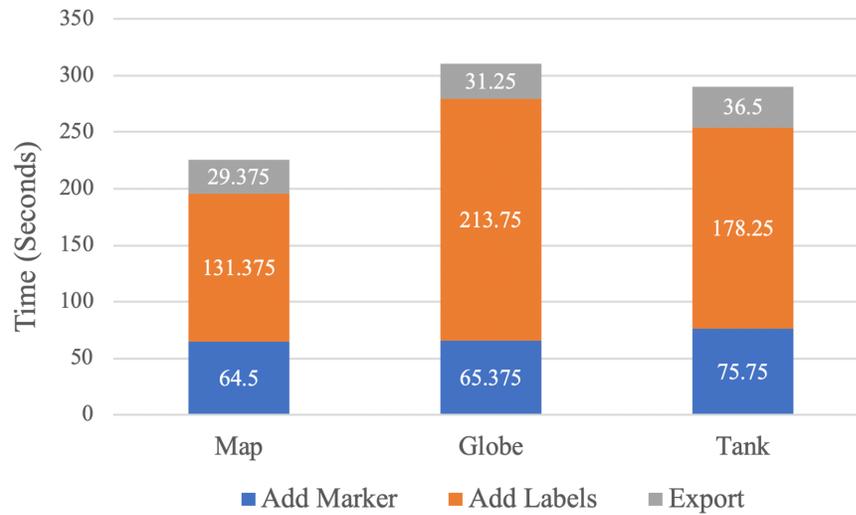


Figure 6.5: The mean time participants spent on each model, broken down by the different sections in Markit.

6.1.4.2 Subjective Feedback

Table 6.1 shows the ten statements from the modified SUS, and the participants' responses. Participants mostly gave positive scores, suggesting Markit was learnable, easy to use, and simple to operate. All participants liked this tool, although some of them had trouble in face selection, object translation, and view navigation when marking elements and moving the tracker scaffold.

Four participants (P1, P3, P4, P5) indicated that Markit could be improved in several ways (relatively negative scores for S2, S3, S5, S7, and S10). P4 and P5 attributed the difficulties they had with 3D navigation to the design of Markit, and P4 emphasized that face selection was a "universal issue" that all 3D modeling software had. As P5 commented, "the fact that you have all these meshes you need to select, just ends up being time consuming." P1 shared the same feeling about the selection functions, but further explained, "the learning curve

1	2	3	4	5	6	7	Mean
S1. I would like to use this tool to create augmented models for blind people when I have time.							5.875
				2	5	1	
S3. The tool is easy to use							5.25
		1		4	2	1	
S7. I would imagine that most people would learn to use this system very quickly.							5.25
		1	1	3	1	2	
S9. I felt very confident using the system.							5.75
				4	2	2	
S2. The tool is unnecessarily complex.							2.25
2	4	1		1			
S4. I would need the support of a technical person to be able to use this system.							2
1	6	1					
S5. Its time consuming to create augmented models with this tool.							2.25
2	4	1		1			
S6. I think the information you can get with this system is too limited.							2.625
1	2	4	1				
S8. The system is cumbersome to use.							2.625
	4	3	1				
S10. I needed to learn a lot of things before I could get along with this system.							2.75
1	2	4		1			

Table 6.1: The ten statements (S1 S10) of the modified Standard Usability Scale (SUS) of Markit's evaluation study, with their histograms and means scores. Each statement was scored from 1 (stands for strongly disagree) to 7 (stands for strongly agree). For positive statements (S1, S3, S7, S9), higher scores are better. For negative statements (S2, S4, S5, S6, S8, S10), lower scores are better.

could be very sharp. Once you grab the sense of how to use different gestures, it could be very easy.” P5 also added, “If you know Blender, I could imagine this could be super easy.”

In addition to a better face selection method, participants expected more output and input options. For example, P8 suggested the Markit should provide indications if the input path or name is incorrect.

6.1.5 Discussion

Markit achieved promising results in the user study. Sighted participants learnt how to use Markit in less than half an hour and added two annotations to a model in 275 seconds (SD = 70) on average.

As participants suggested, better face selection, object translation, and view navigation functions could make Markit easier to use. In the future, we will implement an element detection function, with which Markit could merge faces that belong to an element. For example, the function will detect the faces of “South America” by calculating its boundary and combine them into one item. We will also provide suggestive positions for the tracker scaffold, so users will have fewer translation and navigation operations.

The design of the user study has limitations. The study did not compare Markit with baselines, nor did we evaluate the tool with experts. The main reason is that A3Ms are a relatively new concept and there is no standard tool for creating A3Ms. Thus, it is difficult to define what are the “baselines” and who are the “experts.” The goal of the study was to show the usability of the tool, and the data we collected from amateurs already demonstrated the tool is easy

to use. When A3Ms become more popular in the future, we can design more rigorous comparison-based studies and involve experts to help us improve the model design tools.

Markit enables designers to convert online models into A3Ms. However, Markit still has several limitations. First, it is important to note that most 3D design modeling applications are generally not accessible. Designing 3D models and adding interactive elements with Markit are visually-demanding tasks. Thus, end-users with VIs are still dependent on sighted people, particularly those who have experience with maker technology. Second, it is still difficult to use Markit to do advanced model modification. When creating an A3M, designers may need to add tactile patterns and braille labels. In Markit, a designer would have to use Blender's built-in functions to create a tactile pattern. These functions are commonly used but difficult to operate for non-expert designers.

In the next section, we will describe an accessible design tool for A3Ms. This tool further simplifies the design process of A3Ms, and enables people with VIs to edit A3Ms.

6.2 Molder: A TUI-Based Design Tool

6.2.1 Introduction

Markit presents a universal method to augment printed models with interactive elements. However, Markit only focuses on model augmentation with little support to model modification (e.g., changing tactile patterns on a model). Besides, the GUI of Markit is not accessible to people with VIs. This chapter presents Molder, a more accessible approach that supports model creation, modification, and augmentation. The current design of Molder focuses on augmented printed tactile maps in particular, but it also represents a paradigm for designing other printed tactile materials as well.

Orientation and mobility (O&M) are a major challenge for people with VIs. Many tools have been developed to assist with O&M activities. One important tool is tactile maps, raised-line representations of spaces. Tactile maps are used extensively to help people with VIs learn about a certain area. For example, a TVI or O&M specialist may create a tactile map of a university campus to help a student build a mental map of the campus so she can better navigate to different locations (Espinosa et al., 1998; Kitchin and Freundschuh, 2000).

While tactile maps are very important, they are also difficult to create. TVIs and O&M specialists often create tactile maps for their students with VIs or clients using arts and crafts materials. Such manual creation processes take time and effort and are limited by the maker's artistic ability. Moreover, these maps are not typically robust enough to be reused (Lobben, 2015).

Recently, with advancements in 3D printing, some blind and low vision professionals have begun 3D printing tactile maps and other tactile materials to support accessibility tasks (Buehler et al., 2014). 3D designs of tactile maps can be shared and reused by different professionals. Printed maps are more robust than the ones made by hands and can potentially provide more detailed and accurate tactile features.

Because of the diverse needs of different students or clients, TVIs and O&M specialists need to adjust and customize tactile maps. Student or clients may have different visual and learning abilities, and the maps need to be tailored accordingly. For example, for students who know braille, a TVI may annotate different buildings on a map using tactile patterns and braille labels, but she needs to delete some redundant tactile information for students who are still struggling with the basic concepts of the map

However, most TVIs and O&M specialists are not trained as professional designers, and the design process of tactile maps is still cumbersome for non-expert designers. To design or customize a printable map model, designers need to use modeling software applications (e.g., Rhino (2019)). These applications are not specialized for map design and may require several operations to perform a single task. For example, to add a braille label on a building, a designer needs to create each braille cell one by one, and then position them using complicated spatial navigation operations. In addition, most modeling applications use graphical user interfaces (GUIs). There is also no accessible way for people with VIs to manipulate a 3D object in GUIs, and some O&M specialists and TVIs are visually impaired.

In this section, we present Molder, an accessible map design tool that enables

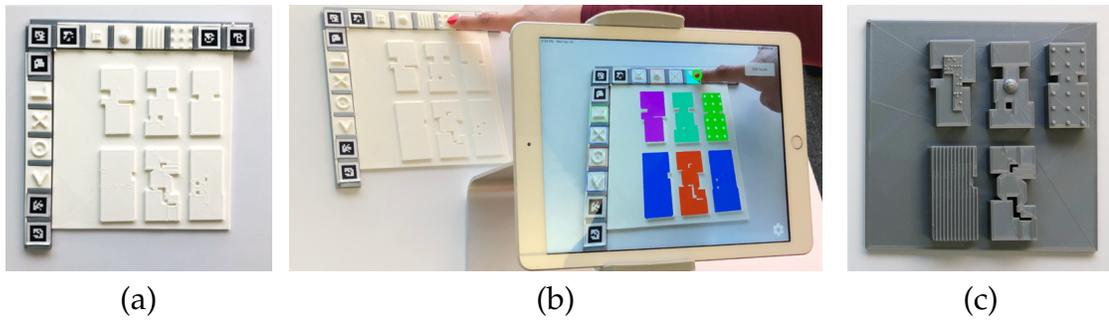


Figure 6.6: A designer is using Molder to modify a map model. (a) He prints a draft map model and places it in the Physical Ruler. (b) Then, the designer uses his finger to select buildings on the printed model and choose desired functions. (c) When he finishes, he prints out the final model, which has a variety of tactile patterns and a bigger size.

non-expert model designers to easily create tactile maps. Molder supports key modeling functions such as adding braille labels, tactile patterns, and interactive labels. All these features are shown be able to make tactile maps more accessible and help people with VIs understand tactile maps better (Brock, 2013; Brock et al., 2015; Giraud et al., 2017; Albuys-Perrois et al., 2018; Ducasse et al., 2018; Holloway et al., 2018). Molder enables a non-expert model designer to (1) create a draft model, add (2) braille labels, (3) tactile patterns, and (4) interactive labels. In addition, Molder also supports (5) element deletion, (6) model scaling, (7) undo function, and (8) interacting with interactive labels.

Molder is easy to use for non-experts. A designer can simply perform all supported functions by physically manipulating a printed draft model and get the information about the model via auditory feedback and high-contrast visual feedback. The designer starts the design process by generating a draft model from Molder. Then, the designer prints out the draft model, and places it in the Physical Ruler, which helps Molder to locate the model, as shown in Figure 6.6a. Instead of performing functions on a 2D GUI on a computer screen, the

designer uses his finger to select buildings from the printed model and choose desired functions from the Tangible Widgets, as shown in Figure 6.6b. Molder senses the designer's gestures and processes the model data in a Blender (2019) server. The final model (Figure 6.6c) can have a variety of tactile patterns and a different size based on the users' modifications. Molder provides visual and auditory feedback for each performed operation, which allows designers with different vision abilities to edit the models. These physical accessories (e.g., the Physical Ruler, and the Tangible Widgets) are reusable and can be printed within hours.

We evaluated Molder in a study with 8 visually impaired non-expert designers and 4 sighted non-expert designers. In the study, the participants finished all design tasks after 30-minute training, and designed tactile map with braille labels, tactile patterns, and auditory labels. On average, the participants spent 22.47 (SD = 18.83) seconds on each modeling task (e.g., adding a braille label). The study showed that Molder was accessible and useful. The participants also suggested ways to make Molder easier to use and learn.

In this section, we will start with the design goals of Molder along with a formative study and a review of prior work, where we tried to understand the current design practice of tactile maps. Then, we will the design and an evaluation study of Molder.

6.2.2 Design Requirements

We derived the design requirements of Molder from a formative study and prior work. First, we conducted a formative study to understand the current design

practice and challenges of creating tactile materials. Second, we revisited prior work to understand how to make tactile maps more accessible by adding tactile patterns and interactive labels. These two steps answered two questions accordingly: How do (expert and non-expert) model designers create accessible models? What kind of design will make tactile maps accessible for people with VIs?

6.2.2.1 The Formative Study

To understand the challenges in the current design process of tactile materials, we joined an online special interest group on 3D tactile materials. The group consisted of more than 30 researchers and practitioners who studied and designed 3D tactile materials for people with VIs, most of them were not professional model designers.

The group members were located worldwide, and they hosted bi-weekly remote group meetings to discuss issues related to 3D tactile materials. Although tactile maps were not the only focus of the group, the discussion of tactile maps emerged with the exploration of 3D tactile materials. We joined three of their remote group meetings, where we learnt their accumulated knowledge and challenges related to 3D tactile materials. In addition to the remote group meetings, we conducted one-on-one interviews with four model designers from the group to further understand their typical design procedure and challenges.

We found that the designers started their design process with a draft model. The draft model is typically downloaded from online resources (e.g., Thingiverse (2019), Touch Mapper (2019)) or converted from a 2D sketch (e.g., using the TactileView software from Thinkable (2019) to convert images to tactile

graphics). Some downloaded and converted models are in stereolithography (STL) format and are therefore difficult to edit in modeling software.

Then, the designers performed several common modeling activities to make the draft model suitable for tactile exploration. They used a variety of software applications like Tinkercad (2019), SketchUp (2019), Blender (2019), and Rhino (2019). Although the software they used was varied, the modeling activities were quite consistent. They usually added braille labels, tactile patterns, texts and symbols to make the tactile materials more understandable for users.

Besides adding additional textures, the designers would also modify the models by resizing them to make sure they are printable and big enough to touch. For overcomplicated models, the designers also deleted and rearranged model components to simplify tactile information.

Although the group members had been using and designing 3D models for a while, most of them were not trained as professional model designers and lacked the skills to do 3D modeling. Thus, they needed to work with a third party to create models. Traditional 3D modeling technique requires long-term training. Moreover, it mostly relies on visual feedback, which is not accessible to designers with VIs. For example, one model designer, who was a low vision teacher, told us that she had to work with a makerspace to produce models and failed to deploy some models in her classrooms due to the long turnaround time.

6.2.2.2 Inspirations from Prior Work

We also reviewed prior work to understand the design practice of accessible tactile maps.

Prior work used braille labels and tactile patterns to annotate elements in tactile maps. In addition to the braille labels (Gual et al., 2011; Götzelmann and Pavkovic, 2014; Kane and Bigham, 2014), there were two types of common tactile patterns. One type was a single primitive like a circular cone and a cube (Gual et al., 2011, 2012). The other type is filling an element with repeated primitives. For example, Taylor et al. (2015) used repeated domes to represent the area of waterways, and filled park area with pyramids.

Researchers also added interactive labels to create interactive tactile maps (Albouys-Perrois et al., 2018; Brock et al., 2011, 2012; Brock, 2013; Brock et al., 2015; Ducasse et al., 2018; Giraud et al., 2017; Holloway et al., 2018). While the driving technologies can be different, the interaction designs of these interactive maps are similar: users touch an element on an interactive map to get its label. These augmented maps enable people with VIs to quickly retrieve rich map information and are accessible to people who do not know braille.

6.2.2.3 The Derived Design Requirements

We derived the following design requirements of Molder based on the findings from the formative study and the prior work:

- Molder should allow designers to create editable draft models.
- Molder should support common modeling functions.

- Molder should be easy to learn and easy to use by non-experts.
- Molder should be accessible to users with VIs.
- Molder should enable designers to add interactive labels.

6.2.3 The Design of Molder

Molder is a tangible modeling tool that enables people to create augmented tactile maps. Molder integrates functions that support a designer to (1) create a draft map model, (2) edit the draft model using common modeling activities, and (3) add interactive labels. When using Molder, a designer performs functions by interacting with tangible objects.

In addition to the tangible objects, Molder has two connected software components: the Molder Application and the Molder Server. Users interact with the Molder Application, while the Molder Server handles model data. The Molder Application is an iOS application that senses the gestures and takes commands, and sends editing commands to the Molder Server, which is a server that modifies the model accordingly. Then, the Molder Application renders auditory and visual feedback to the designer.

In this section, we will first walk through the tangible objects and the functions of Molder. Then, we present the design and implementation of the software components.

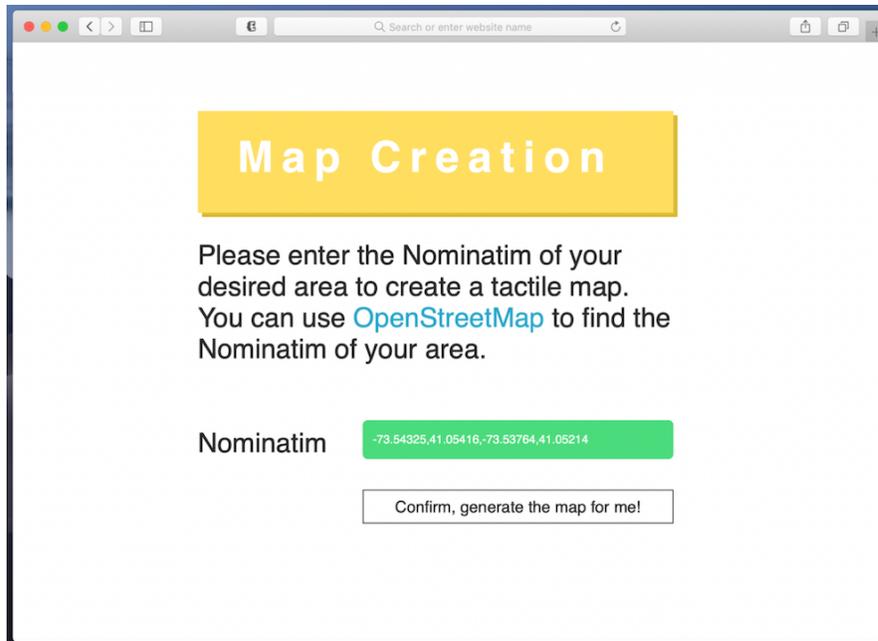


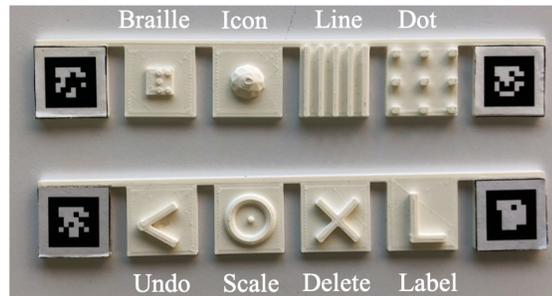
Figure 6.7: The user interface of the map creation web page.

6.2.3.1 Tangible Objects

Molder has four types of printed tangible objects: a printed draft model, the Tangible Widgets, the Physical Ruler, and the Indicator.

Draft Models

A designer can create a draft model by simply entering the geographic information of an area in a web page, as shown in Figure 6.7. The designer can obtain the geographic information (i.e., Nominatim) by entering an address or a zip code in OpenStreetMap (2019). Molder automatically generates the draft model using the geographic information. Then, the designer prints out the draft model using a 3D printer and places the printed model to the Physical Ruler. Each draft model has a fixed size of $160 \times 160 \times 4 \text{ mm}^3$.



(a) There are two tangible widgets. The Tactile Widget has four functions: Braille, Icon, Line, and Dot. The Function Widget consists of functions like Undo, Scale, Delete, and Label.



(b) Designers can combine the two Tangible Widgets into a palette.

Figure 6.8: The design of the Tangible Widgets.

The Tangible Widgets

We designed two Tangible Widgets: the Tactile Widget and the Function Widget. Both consist of tactile representations of different functions, as shown in Figure 6.8. The Tactile Widget is dedicated for adding tactile features, and has four functions: Braille, Dot, Line, and Icon. Function Widget has four other functions: Label, Delete, Scale, and Undo. Each Tangible Widget has two markers on its two sides, which enable the Molder Application to locate their positions. Designers can put the Tangible Widgets on the Physical Ruler or combine them into a palette.

The Physical Ruler

The Physical Ruler is made of two orthogonal rulers. Each ruler has six empty slots for the Tangible Widgets. There are three markers on the Physical Ruler and a stopper for the Indicator (Figure 6.9).

The Indicator

The Indicator is used in the Scale function and is a rectangle that has trackable markers (Figure 6.9). The right side of the Indicator has a rigid tactile pattern to show its orientation. There are two stoppers on the Indicator that can work with the Physical Ruler to demonstrate the original width of the printed draft model (i.e., 160 mm width).

6.2.3.2 Functions

Model Exploration

The designer can explore the printed model through visual and auditory feedback. The designer puts a red sticker on one of her fingers. The sticker allows Molder to track the position of the finger. Each buildings of the draft model have an assigned numeric ID to help the designer understand the model. When the designer briefly touches on a building, Molder will report the ID of the building. She can hold her finger on the building to select the building and get the detailed design information of the building. For example, when she selects the building with an ID of 2, Molder says “2 selected, has a braille of ‘library’, no label, no pattern,” which indicates that the building has a braille label but does not have any audio label and tactile pattern. Molder also provides high-contrast

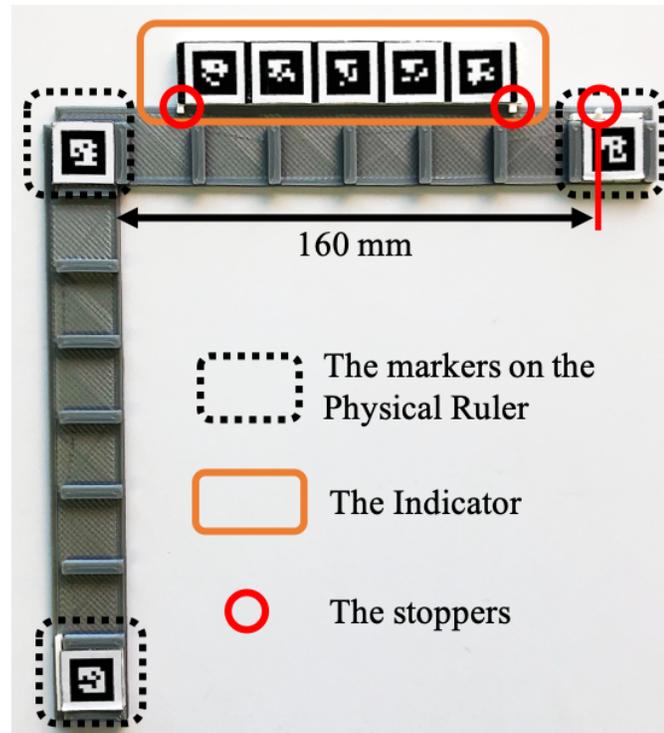


Figure 6.9: The Physical Ruler and the Indicator. There are three markers and one stopper on the Physical Ruler. The Indicator has two stoppers and a rigid tactile pattern on its right side.

visual feedback for low-vision designers.

Deletion

If the draft model contains unnecessary buildings, the designer can delete these buildings. She selects an unnecessary building, and then selects the Delete function on the Function Widget. Molder provides auditory and visual feedback to inform the designer that her command has been executed and updates the model information. Because the building still exists on the printed draft model, Molder will speak “no element, already deleted” when the designer tries to select the deleted building again.

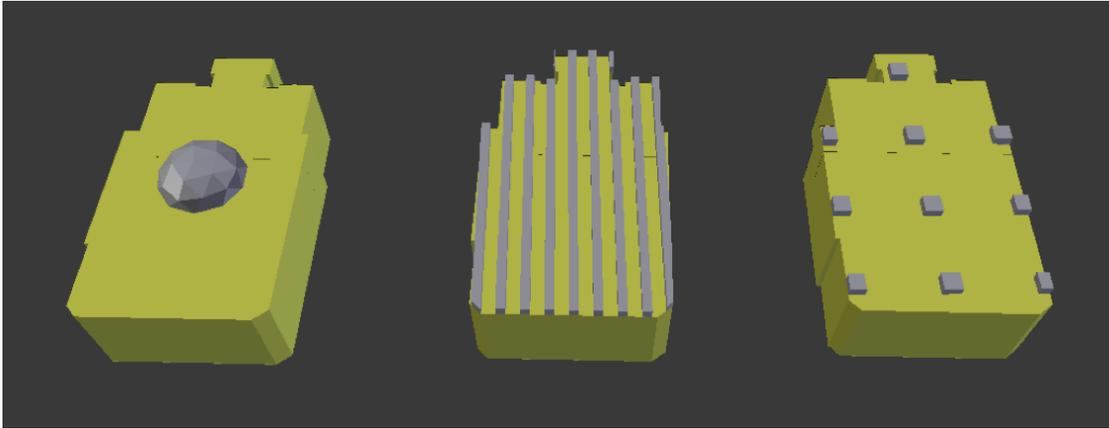


Figure 6.10: Three tactile patterns on a same building. From left to right, the added tactile patterns are Icon, Line, and Dot. The building is marked in yellow and the added patterns are marked in grey.

Tactile Patterns

The designer can assign a tactile pattern to a building. There are three types of tactile patterns: Icon, Line, and Dot. We chose these patterns based on the formative study and prior work. The designer selects a building and chooses one pattern to apply on the building. Then, Molder adds the pattern on the selected building. If the designer is unsatisfied with the current tactile pattern, she can overwrite the current pattern by applying another one. Figure 6.10 shows the three patterns on a same building.

Braille Labels

The designer can add a braille label to a building. She selects a building and activates the Braille function on the Tactile Widget. Then, she enters the text of the braille label in a pop-up textbox. On iOS, the designer can use VoiceOver's dictation function to enter texts (Figure 6.11a). Then, Molder processes the com-

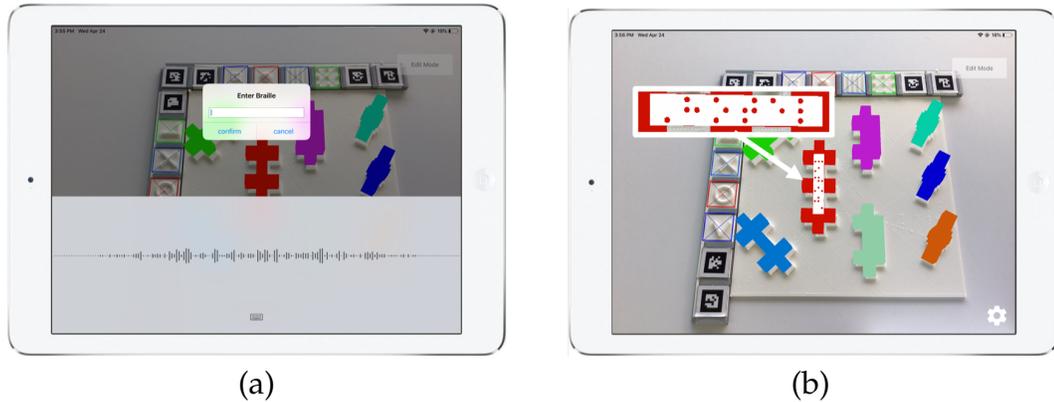


Figure 6.11: To add a braille label, a designer selects a building and the Braille function, and (a) enter texts on a popup textbox. (b) Molder processes the command and update the model. The braille label entered here is “Hotel.”

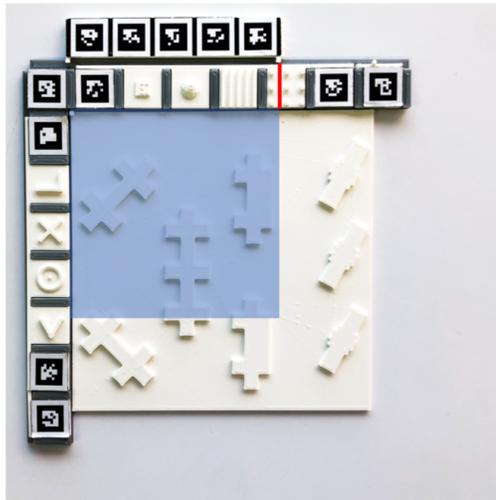
mand and update the model (Figure 6.11b). The designer can overwrite the current braille label with another one.

Interactive Labels

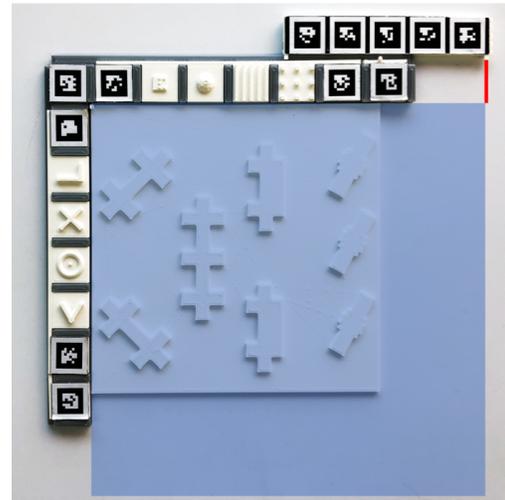
In addition to tangible braille labels, the designer can also add virtual audio labels. Unlike a braille label that is limited by its size, interactive labels can contain more information. To add an interactive label, the designer needs to select a building and activates the Label function on the Function Widget. Then, she enters texts in a pop-up textbox. As similar to adding a braille label, the designer can repeat this process to overwrite an audio label.

Scaling

The designer can resize the model using the Indicator. She puts the Indicator on the top of the Physical Ruler. The right end of the Indicator stands for the final width of the model. To make the model smaller, she moves the Indicator to left (Figure 6.12a). To make the model bigger, she moves the Indicator to right



(a) To make the model smaller, the designer moves the Indicator to left.



(b) To make the model bigger, the designer moves the Indicator to right.

Figure 6.12: A designer can use the Indicator to change the size of a model. The red lines in the images stand for the right end of the Indicator. The blue rectangles represent the final sizes of the model.

(Figure 6.12b). Then, she holds on the Scale function to apply the scale. The Scale function is also overwritable. While the model data will be updated in the server (i.e., the Model Server), the model on the iOS application still keeps the original size in order to align with the printed draft model to avoid confusion.

Undo

If the designer makes a mistake, she can either perform an overwrite command or execute the Undo function. Functions like Braille, Label, Scale, and the tactile patterns are overwritable. The designer can overwrite the content with a different value (e.g., same building with a different braille label). The designer can also use the Undo function to cancel the command that is performed most recently. For example, if the designer accidentally deletes building 3, she can hold the Undo function to bring the building back.

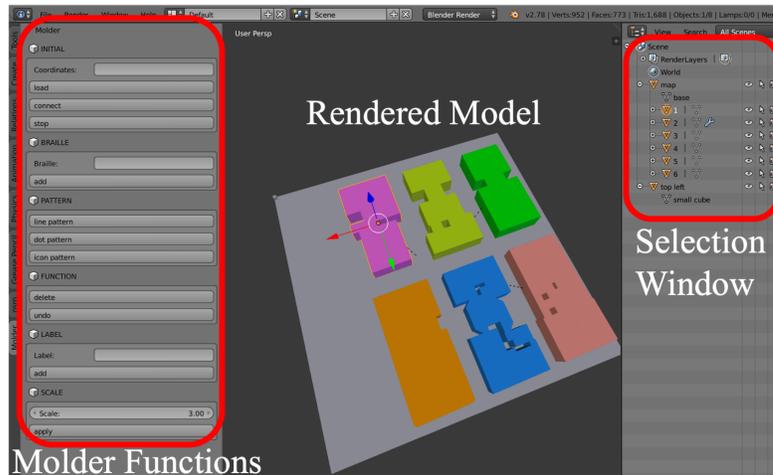


Figure 6.13: The user interface of the Molder Server. A user can select a building in the selection window and perform Molder functions. The model will be updated in real time.

Interaction Mode

Once the designer finishes her edits, she can print the new model for her students. Molder has an interaction mode that allows students to explore the model with its tactile features and audio labels.

6.2.3.3 The Molder Server

The Molder Server is in charge of processing model data, and is implemented as an add-on in Blender (2019), a modeling software that includes basic modeling functions (e.g., Transformation, Boolean Modifiers). Blender provides Python APIs to its built-in functions, which allowed us to edit 3D models using Python scripts. The user interface of the Molder Server is shown in Figure 6.13.

Communication

The Molder Server is a server that takes commands from different user interfaces. Although the Molder Server can be used as a standalone design tool, its user interface (i.e., Blender GUI) is not accessible, and is not compatible with screen readers. Thus, we implemented it as a server, which can take commands from server clients. In the current design, we implemented two accessible user interfaces as server clients: the web page of map creation and the Molder Application. Designers use the web page to create map models, and edit the model in the Molder Application, an iOS application. The two accessible user interfaces send commands (e.g., creating a map with geographic information) to the server, and receive updated data (e.g., a printable file for a draft model) from the server.

Model Creation

The Molder Server uses Blender OSM (vvoovv, 2019) to pull map data from OpenStreetMap (2019). Blender OSM provides Python APIs that enable us to import OpenStreetMap data using geographic information. Molder only imports 2D building outlines using Blender OSM (Figure 6.14a).

Then, the Molder Server applies different operations to create a draft map model using the 2D building outlines. The Molder Server combines overlapping units. OpenStreetMap may register a single building as several overlapping units (e.g., a building with a dozen shops). As our goal is to provide a map for an area instead of a single building, we combine these overlapping units to avoid overcomplicated information. The Molder Server also assigns high-contrast colors and IDs to each building to distinguish them (Figure 6.14b).

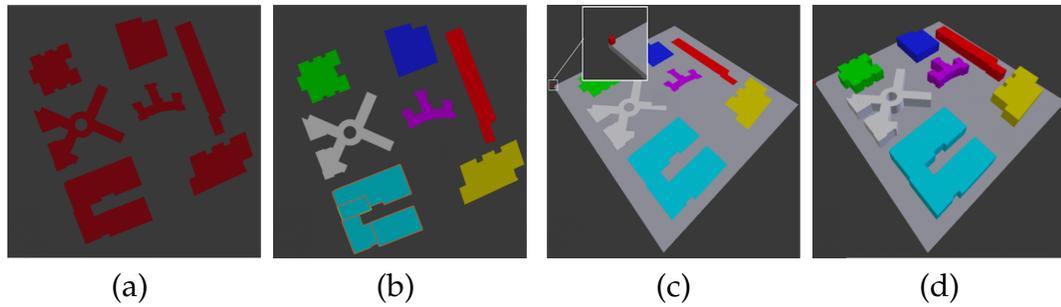


Figure 6.14: Molder automatically creates a draft model with the geographic information provided from a designer. (a) First, Molder imports 2D building outlines. (b) Then, it combines overlapping units to a single building (e.g., the aqua building has four units) and assigns high-contrast colors and IDs to all buildings. (c) Molder also adds a base (marked in grey) and a small cube (marked in red) on the left top corner to indicate the orientation of the model. (d) Last, Molder extrudes the 2D building outlines to 3D meshes.

Third, the Molder Server creates a rectangle base for the map model. The base can fit into the Physical Ruler, which aligns a digital model to its printed model. The dimension of the base is $160 \times 160 \times 2 \text{ mm}^3$. We also add a $2 \times 2 \times 2 \text{ mm}^3$ cube on the left top corner of the base to indicate the orientation of the model (Figure 6.14c). When creating the base, we also align the model according to its bounding box and resize the model to fit into the base.

Last, the Molder Server extrudes the 2D building outlines to 3D meshes with different heights and creates two versions of draft models. One shallow draft model has buildings with 2 mm height, and another regular draft model has buildings with 10 mm height. The shallow draft model is faster to print, and the regular draft model serve as an unedited model. Figure 6.14d shows the regular draft model. All edits will be performed on the regular draft model. We set these heights according to our experience with tactile materials, in the future, designers should be able to customize the height of the regular model. Figure

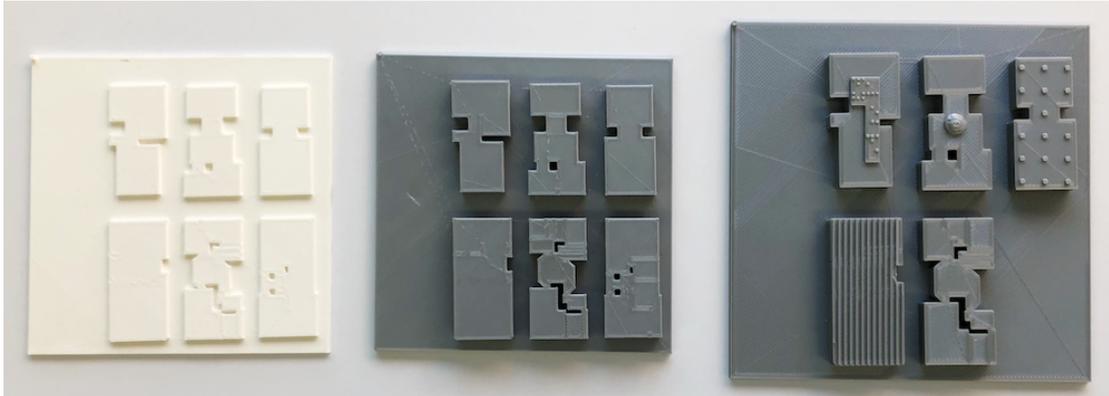


Figure 6.15: From left to right: a shallow draft model, an unedited regular draft model, and a final model.

6.15 shows a shallow draft model, an unedited regular draft model, and a final model.

Deletion

Each building is an independent mesh, and we use Blender’s functions to delete the one that a designer selects.

Tactile Patterns

For the Icon pattern, the Molder Server creates an icosphere at the center of a selected building. For the Line pattern and the Dot pattern, we fill the bounding box of the selected building with rectangle or dot meshes and perform an intersect Boolean modifier to trim all meshes into the shape of the selected building.

Braille Labels

The Molder Server converts the text input from a designer to a braille label on a selected building. In the current design, we only implemented grade 1 braille, where each cell represents only one letter, number, punctuation sign, or special

Braille composition sign. The Molder Server converts English characters into a printable braille label using similar algorithms implemented in braille-printer (louisheath, 2017). Then, the printable braille label is moved to the center of the selected building.

Audio Labels

The Molder Server stores audio labels as self-defined objects.

Scaling

The Molder Server resize the entire map model with a designer-define value. When resizing the model, the Molder Server keeps the original size of braille labels since braille has a standard size.

Undo

The Molder Server registers each operation with Blender, which allows us to perform the Undo function using Blender's built-in return mechanism.

6.2.3.4 The Molder Application

The Molder Application is an iOS application that uses computer vision to locate the positions of the tangible objects and tracks finger gestures. The Molder Application is also in charge of rendering feedback to a model designer. In addition, end users (e.g., students with VIs) can also use this application to retrieve the audio labels on models. The algorithms and implementation of the application are similar to those in Talkit and Talkit++, as described in Section 4.2.3.2.

Locating Tangible Objects

The Molder Application relies on markers to locate tangible objects including the printed draft model, the Physical Ruler, the Tangible Widgets, and the Indicator. We use the Chilitags library (Bonnard et al., 2013) to locate the markers on the tangible objects. Once locating the markers, the application uses spatial information and transformation matrices to determine the positions of these tangible objects.

Tracking Finger Gestures

The Molder Application tracks the finger by recognizing the red sticker on a designer's fingernail. Then, the application compares the position of the tangible objects and the finger. To explore different elements (e.g., buildings and functions), a designer can briefly touch an element to get its name (e.g., the name of a function or the ID of a building).

The Molder Application uses time as a variable to recognize different selection. There are two types of selection: selecting a building and selecting a function. To select a building on a draft model, the designer needs to hold her finger on the building for 1.5 seconds. The application will report the detailed design information of the selected building. The selected building will be timeout and deselected after 10 seconds if there is no further modeling activity. To select a function, the designer needs to hold her finger for 3.5 seconds. The application speaks "hold to apply" after 2 seconds, and then "applied" after 3.5 seconds. Because applying a function by mistake will require several steps to fix, we set the time of selecting a function to be longer than the time of selecting a building. We set these thresholds empirically and intended to help designers avoid

accidental operations. In the future, researchers can also consider combining gestures and speech commands to create more intuitive interactions (Bolt, 1980; Geißler, 1995).

Rendering Feedback

The Molder Application provides visual and auditory feedback to help designers understand the current modeling status. For visual feedback, the application highlights different functions and buildings. In addition, it renders different tactile patterns on the buildings. The application also adds a flashing animation on a selected building. For auditory feedback, the application speaks all key steps in the modeling process (e.g., selecting a building, performing a function). The designer can also query the detailed design information of a building by selecting it. The application also notifies designers when the tangible objects are out of the camera's view.

Switching Modes

A student can use the interaction mode of Molder by tapping a button on the top right corner of the application. In the interaction mode, the student can briefly touch a building to retrieve its audio label. The application pulls data from Molder Server and renders audio output.

6.2.4 Evaluation

We evaluated the usability and usefulness of Molder with two research questions: (1) Can people with VIs use Molder to modify models? (2) How do sighted and visually impaired model designers think about the implemented

functions and interactions?

6.2.4.1 Method

Participants

We recruited 12 participants (2 males, 9 females, 1 other mean age = 57.5, SD = 16). All participants had experience modifying and designing tactile maps. We recruited participants through fliers at Low Vision community centers and email blasts through schools for TVIs.

Eight participants with VIs, denoted as P1- P8, joined the study. Four participants identified as blind (P3-P5 and P8) and four participants identified as low vision (P1, P2, P6 and P7). The participants had experience using and designing tactile materials. Most of them acquired the design experience by helping other designers to create tactile materials like tactile maps of subway systems. Only P4 had used a 3D modeling tool (i.e., OpenSCAD (2019), a script-based 3D modeler) before the study. All of them were familiar with braille, and seven participants were familiar with VoiceOver. None of them had enough vision to use the GUI of Molder Server.

In addition, four sighted participants (all females) joined our study. These participants had experience teaching and creating tactile materials. Three participants were working or have worked as TVIs while the remaining one (P10) has designed tactile graphics before. Only P10 used a 3D modeling tool (i.e., OpenSCAD) before the study. All of them were familiar with braille, and VoiceOver.



Figure 6.16: Three draft models we used in the study. From left to right: Model A, Model B, and Model C.

Apparatus

We set up the Molder Server on a Macbook and ran the Molder Application on an iPad. The iPad was held in front of a participant by a stand (as shown in Figure 6.6b). We used the back camera of the iPad to track objects and added a wide-angle lens to enhance the camera's field of view.

We prepared three draft models, denoted as Model A - Model C (Figure 6.16). Model A has 6 buildings, Model B has 8 buildings, and Model C has 12 buildings. We also printed a modified model of Model A to demonstrate different functions. Model A and its modified model can also be found in Figure 6.15. In order to let our participants better understand the models they would design, we chose these three models to represent three different local neighbors.

Procedure

The study consisted of a single session that lasted around 1.5 hours. First, we introduced Molder to participants. The participants learned how to use the tangible user interface (TUI), which includes different tangible objects, functions,

and the Molder Application. For sighted participants, we also introduced how to use the GUI (i.e., the Molder Server) as a standalone design tool. We used Model A and its modified model in introduction. This process took around 30 minutes. Then the participants performed two tests:

- *Test 1: Structured Design.* The participants modified Model B according to our instructions. In this test, there were nine tasks, denoted as T1 - T9, as shown in Figure 6.17. These tasks, which covered all the functions of Molder, were chosen based on the common modeling activities we found in the formative study. We announced the tasks one by one. The participants performed the announced task and spoke “done” when they finished the task. participants with VIs only used the TUI, while sighted participants used both TUI and GUI to perform the same tasks on Model B twice. We counter balanced the order of two interfaces for sighted participants.
- *Test 2: Freelance Design.* The participants modified Model C in this test. They could design the model in their own ways. Participants with VIs used the TUI, while sighted participants chose one of two interfaces.

Last, we wrapped up the study with a short interview, where we used modified Standard Usability Scale (SUS) (Brooke, 1996) to evaluate the user experience of the TUI. The modified scale has ten statements, and all participants were instructed to rate the statements for the TUI. We asked participants to rate the statements on a scale from 1 to 7, where 1 was “strongly disagree” and 7 was “strongly agree.” To reduce acquiescence and response biases, we mixed positive and negative statements.

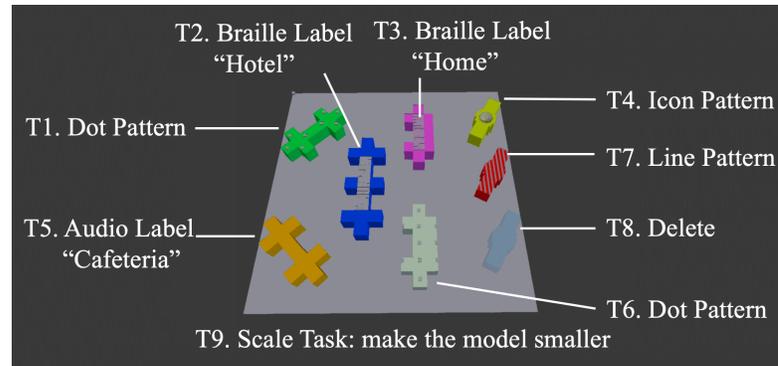


Figure 6.17: The nine tasks in the study, including two braille labels (T2, T3), one audio label (T5), four tactile patterns (T1, T4, T6, and T7), one deletion (T8), and one scale task (T9).

Data Collection and Analysis

We video recorded all sessions and analyzed the videos. We calculated the time participants spent on each task in the structured design test. We also analyzed the modeling activities they had in the freelance design test. We also transcribed participants' comments and identified common themes in the final interview.

6.2.4.2 Results

The Time Participants Spent in the Structured Design Test

All participants managed to finish T1 - T9 using assigned user interfaces. P3, self-identified as blind, had no experience with VoiceOver. Thus, we performed the VoiceOver dictation gesture (i.e., two-finger double tap) for her when she needed to enter text. The other participants finished all tasks independently. We plotted the time each participant spent on the tasks in Figure 6.18.

Across all participants and user interfaces, participants spent 22.47 (SD = 18.83) seconds on each task. On TUI, participants with VIs (i.e., P1 - P8) spent

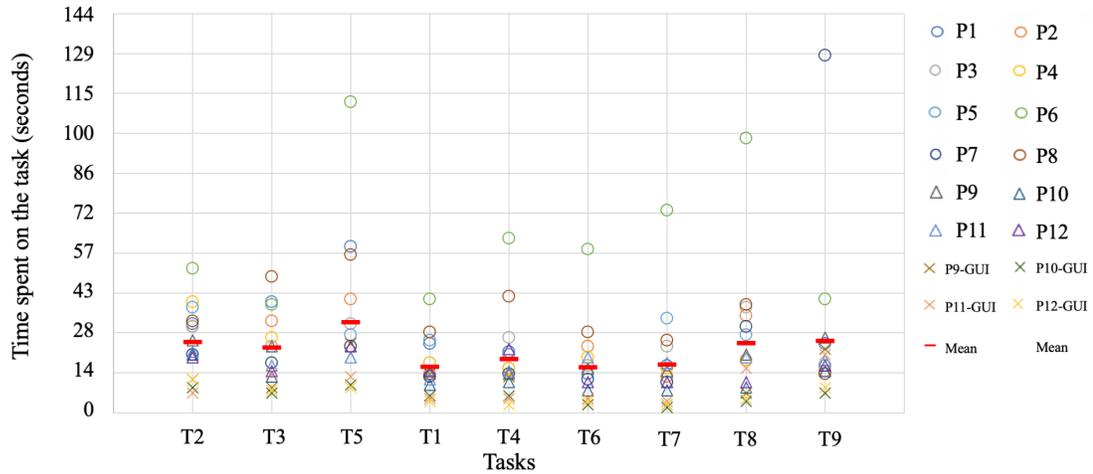


Figure 6.18: The time participants spent on each task. T2, T3, and T5 required text entry. T1, T4, T6, and T7 are related to different tactile patterns. T8 is a deletion task and T9 is a scaling task.

32.18 (SD = 21.83) seconds on each task, while sighted participants spent 16.83 (SD = 5.27) seconds on each task. On GUI, sighted participants spent 8.67 (SD = 5.73) seconds on each task.

On average, participants with VIs used more time compared to sighted participants. We ran an ANOVA analysis, which considered the type of tasks (i.e., T1 - T9) and the eye conditions (i.e., visually impaired and sighted) as variables. On average, we found the participants with VIs spent 15.35 seconds more than sighted participants ($p\text{-value} = 5.5 \times 10^{-5}$). We found participants with VIs needed to find buildings and functions through tactile exploration, while sighted participants could quickly locate their desired objects. The time spent by participants with VIs was also more varied (higher standard deviation). We think this may relate to their different visual abilities. Among all participants with VIs, P6 spent the longest time on average (mean = 60.7 seconds, SD = 26.13). She wore long acrylic nails and had difficulty selecting objects. P7

spent 129 seconds on T9 because she forgot where the Scale function was.

Sighted participants spent less time on GUI compared to their performance on TUI. We ran another ANOVA analysis, which took the type of tasks (i.e., T1 - T9) and the interfaces (i.e., GUI and TUI) as variables. We found that sighted participants spent significantly more time on TUI compared to GUI (p -value = 7.1×10^{-5}). On TUI, participants needed to hold their fingers to select an object. The time thresholds for selecting a building and a function are 1.5 seconds and 3.5 seconds respectively. While on GUI, they selected an object without a time threshold.

The Modeling Activities in the Freelance Design Test

In Figure 6.19, we plotted the count of the modeling activities in the freelance test. P1 - P11 used the TUI, and P12 used the GUI. On average, each participant with VIs did 9.75 (SD = 3.69) activities, and each sighted participant did 13.5 (SD = 2.38) activities.

All participants used braille labels and nine participants used audio labels. As for content, they used general terms (e.g., “church”) and specific names (e.g., the name of a bar). The three participants who did not use audio labels (P6, P10, and P11) were not sure about the use cases of audio labels. Other participants showed different preference between braille labels and audio labels. P1 thought “braille labels are for more important things” and audio labels were for orientation purposes. Four participants (P2, P3, P8, and P9) said their choices were arbitrary because they wanted to try both labels out. Two participants (P4 and P5) would like to add the two types of labels with same content to keep consistency. P11 and P12 preferred audio labels because “not everyone would know

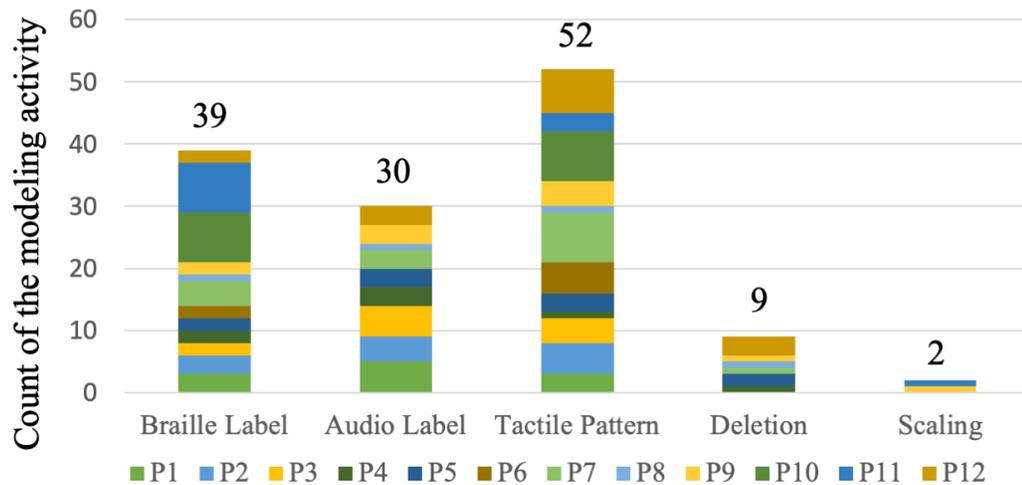


Figure 6.19: The count of the modeling activities in the freelance test.

how to use braille.”

Subjective Feedback

Overall, participants highly rated Molder. Table 6.2 shows the responses to the modified SUS. The participants thought Molder could help them design tactile maps (positive scores for S1 and S5). However, P10 thought the TUI is less useful compared to the GUI (relatively negative scores for S1 and S5) and said, “It’s fun, but it wouldn’t make it faster.” Most participants thought they could handle the system by themselves (positive scores for S2, S4, S9). P3 gave a negative score for S2 because she received our help for text entry tasks. P6 thought she might need assistance when learning the system.

The participants managed to learn Molder and performed different tasks in short time, but they thought the learning curve of Molder could be stiff (relatively negative scores for S7 and S10). Although the participants had experience

1	2	3	4	5	6	7	Mean
S1. I would like to use this tool to design tactile maps.							5.92
1			1	1	2	7	
S3. The system is easy to use							5.75
			3	1	4	4	
S5. I think the system will help me design tactile maps.							6.5
		1			2	9	
S7. I would imagine that most people would learn to use this system very quickly.							5.42
		1	2	3	3	3	
S9. I felt very confident using the system.							6.33
				1	6	5	
S2. Its difficult to handle the system by myself.							2.08
4	5	1	2				
S4. I would need the support of a technical person to be able to use this system.							1.92
4	6	1	1				
S6. I think the functions of this system are too limited.							2.67
4	4		4				
S8. The system is cumbersome to use.							2.01
5	4	1	1	1			
S10. I needed to learn a lot of things before I could get along with this system.							3.08
2	5	1	1		3		

Table 6.2: The ten statements (S1 S10) of the modified Standard Usability Scale (SUS) of Molder’s evaluation study, with their histograms and means scores. Each statement was scored from 1 (stands for strongly disagree) to 7 (stands for strongly agree). For positive statements (S1, S3, S5, S7, S9), higher scores are better. For negative statements (S2, S4, S6, S8, S10), lower scores are better.

in design and tactile materials, modeling is still a new task for many participants. P5 said, “I wouldn’t say that it’s necessarily hard, just different because it’s new,” which was echoed by P2, P3, P9, and P12. P6 and P12 also showed concerns for visually impaired people with neuropathy, who might not be able to use Molder because they are losing tactile sensation.

Some participants thought Molder could have more functions (relatively

negative scores for S6). For example, P4, P8, and P9 wanted to have more choices for tactile patterns. P2 and P8 would like to add different layers of audio labels so users can get more detailed information if they want. P2 and P7 wanted to add new objects (e.g., stairs, benches) on draft models. In addition to braille labels, P10 also wanted printed texts, which can be used by people who do not know braille.

6.2.5 Discussion

Molder enables non-expert designers to create and modify tactile maps on tangible user interfaces. In the study, all participants managed to finish tasks and showed their interests in using Molder for future designs. The study also exposed some limitations of Molder, which lead to our thoughts on future work.

6.2.5.1 Tangible Modeling

In the study, the participants expressed their excitements about Molder and tangible modeling. Four participants with VIs (P1, P2, P4, P8) specifically mentioned they liked the feelings of “just creating different things” by touching the models. P10, a sighted participant, pointed out that the TUI makes it easier to envision the final design and said, “I would imagine to not be staring at a computer screen, I like that I’m looking at a physical surface.”

In the study, both participants with VIs and sighted participants could use the TUI of Molder to finish modeling tasks, although participants with VIs spent more time as they needed to find objects through tactile exploration. In the current implementation, Molder only supports selection using one finger, which

limits the tactile exploration process. In the future, we will consider two-hand multi-finger interactions.

For sighted participants, the TUI is slower compared to the GUI due to the time thresholds we set. In the future, we should enable users to customize the time thresholds.

6.2.5.2 Learnability

In the study, the researchers taught the participants how to use Molder within a half hour. Some participants found it challenging to master Molder in the limited time. P8 suggested that a learning mode would be more sustainable and could improve the learnability of Molder. In the future, we should deploy a tutoring application in Molder, which enables designers to learn how to modify models independently in long term.

6.2.5.3 Fabricating Draft Models

To use Molder, a designer must fabricate a draft model first. In the study, we fabricate the models using 3D printers. The printing process is time-consuming. It takes several hours to print a shallow draft model. As alternatives, designers can use swell form machines and tactile graphics embossers to fabricate a draft model, which only take several minutes.

6.2.5.4 Extending to Other Models

Molder is designed for tactile maps but can be potentially used to design other types of 3D models. To modify a 3D model in Molder, the 3D model has to be segmented. In current implementation, all buildings on tactile maps are segmented into independent objects. To segment other types of 3D models, we can use automatic segmentation algorithms (e.g., the one proposed by Koschan (2003)) or use crowd-powered systems. For example, a designer with VIs can download a 3D globe model online. Then, she can upload to a crowd-powered system to preprocess the model. Crowd workers can help the designer select and segment out different continents.

6.3 Summary

In this chapter, we introduced two design tools for A3Ms: Markit and Molder.

Markit, which is implemented as a Blender add-on, enables designers to easily add audio annotations to online models. To create an A3M, a designer downloads a model online, and imports it into Markit. She adds a tracker scaffold to the model and associates text with areas on the model. Markit can handle different types of models but its GUI is not accessible to designers with VIs.

On the other hand, Molder is specialized at tactile maps and can be used by designers with different vision abilities. Molder is a TUI-based design tool for tactile maps. A designer can use Molder to create and modify a tactile map model. The designer modifies the model by physically manipulating the model. Molder provides auditory feedback and high-contrast visual feedback to help the designer in the design process. Molder is accessible to designers with different visual abilities and user-friendly to non-expert designers

Markit and Molder present two interaction paradigms of model design. Markit is based on a GUI, which is widely adopted by commercialized modeling software applications. Molder reinvented the modeling process using a TUI, which allows users to edit a model using tangible objects, gestures, and speech input.

CHAPTER 7

DISCUSSION AND FUTURE WORK

In the previous chapters, we have explored different sensing technologies, design studies, and design tools of A3Ms. This chapter will discuss the findings from our prior work and propose future work.

Eventually, we hope to foster an online community that provides A3Ms to people with VIs. We envision that people with VIs will be able to access 3D printers in their daily lives and print models easily. Model designers can easily add interactive information onto 3D models, and users with VIs can download the data along with the digital file representing the 3D model. This process involves several interconnected components: the design of tactile features, the fabrication of tactile learning materials, the design of interactive elements, and the driving sensing technologies. The content of this chapter will be divided into these interconnected components.

7.1 The Design of Tactile Features

Before augmenting a model, a designer needs to design and modify the tactile features of the model first. In Section 5.4, we discussed the design guidelines of the tactile features of A3Ms. In Chapter 6, we demonstrated two design tools that can modify the tactile features of a model. Moving forward, we need to improve the design tools to further support the design guidelines.

One design guideline from Section 5.4 implies that tactile features should be simple, clear, and salient, and Molder complies with this design guideline.

For example, when designing a tactile map for navigation purpose, it may not be necessary to demonstrate different heights of the buildings on the map. To make the tactile features on map models simple and salient, Molder, the design tool for augmented tactile maps, simplifies map models and allows designers to delete redundant buildings.

In the future, we need to create new functions that can create simple and salient tactile features on other types of models. There are computational methods to simplify models (Cohen et al., 1997). However, the tactile features that are eliminated by these methods could be the ones that students need to learn. For example, when simplifying a dragon model, these methods tend to simplify dragon scales (Lindstrom, 2000), which could be a key tactile feature that a student want to know. Thus, we need to involve the opinions from designers and students. For example, the design tool can have generative design functions, which can generate multiple design candidates. Designers can choose one of these candidates or change parameters to generate other design candidates.

Another design guideline suggests that we should use removable components to allow users to explore the relationships between a model's different components. Creating removable components in commercialized modeling software applications usually requires professional design skills. A designer needs to segment the removable components and adjust the gap between the components and the main body. This process is also visually-demanding. In the future, we can create new design tools based on TUIs. For example, a designer can use clay to create different removable components and scan them into a design tool. The design tool will further process the raw model data and adjust the sizes of different components automatically to make sure they can fit together.

7.2 The Fabrication of Tactile Learning Materials

With current consumer-grade 3D printing technologies, printed tactile learning materials still have limitations in printing quality and model designs. Most printed models have printing imperfections and will not feel the same as the referent objects. In addition, it still takes a nontrivial amount of time to design, print, and prepare annotations for models. More importantly, some printers may not be accessible enough for people with VIs to operate. For example, most printers have physical buttons and digital displays that do not have built-in screen readers. While on-demand printing services could help people deal with the inaccessible printing process for now, we think it is important to improve the accessibility and user experience of the fabrication process in the future.

In the future, we need to investigate faster fabrication technologies that are more suitable for printed learning materials. Tactile learning materials have special features compared to other printed objects. First, learning materials may have more tactile features and textures. To fabricate learning materials with complex tactile features and texture, we can combine different materials and fabrication methods. For example, Peng et al. (2015) combined 3D printing technologies with laser cutting technologies to fabricate soft objects. This method can potentially be extended with other materials to create objects that share similar tactile features with their referent objects.

Additionally, tactile learning materials do not need to be as robust as functional prototypes (e.g., a printed handle). Thus, we can speed up the fabrication process of tactile learning materials using new printing algorithms. Since the robustness of a tactile learning material is not our priority, we can fabricate an

object with minimal printing infill. For example, Mueller et al. (2014) printed the wireframe preview of an object to speed up the prototyping process.

7.3 Sensing Technologies

This dissertation discussed two types of sensing technologies for A3Ms: acoustic sensing (Section 4.1) and RGB-based computer vision (Section 4.2). As more sensors become mainstream, we can develop new sensing technologies for A3Ms.

Dual-camera systems and depth cameras are more available in mobile devices. The depth information from these camera systems can help us locate models a visual tracker. In prior work, researchers used depth cameras to track texture-less objects (Hinterstoisser et al., 2012). It is also possible to build systems that learn the features of a model and track it without a prior digital model (Sharp et al., 2015). These techniques can be used for tracking A3Ms as well. In addition, depth information allows us to track more complicated gestures without a visual tracker (Qian et al., 2014; Sharp et al., 2015). In the future, we will explore how to apply these methods to A3Ms. The gestures proposed in Section 5.1 can also be used as a reference when designing new gestures.

7.4 The Design of Interactive Elements

Markit and Molder supports designers to add interactive elements on 3D printed tactile materials. Using Markit, a designer can select a component on a model, and associates it with textual information and links to multimedia ma-

terials (e.g., audio recordings, short videos). The current Molder only supports simple textual information but can be extended with multimedia files.

However, it is still challenging to design interactive elements. While textual information can be derived from textbooks, audio recordings and short videos are not easy to create. These multimedia learning materials can help students better understand a learning material, but TVIs and O&M specialists may not have skills to design these multimedia materials. In our projects, we helped TVIs to design multimedia materials using professional tools like Final Cut Pro from (Apple, 2019c) and Adobe Creative Cloud (Adobe, 2019). However, these tools have steep learning curves. In the future, we need to explore easy-to-use tools for creating interactive elements.

CHAPTER 8

CONCLUSION

This dissertation introduces augmented printed models (A3Ms), interactive tactile learning materials. These A3Ms are designed to help students with visual impairments (VIs) to learn visual concepts (e.g., the structure of a DNA model).

We started with different sensing technologies for A3Ms. Initially, we designed Tickers and Talker, an acoustic sensing-based toolkit for 3D models. Tickers are small percussion instruments that generate unique sounds when strummed. Talker is a signal processing application that uses a microphone to sense and classify Ticker sounds, and speaks the label corresponding to the strummed Ticker. We evaluated Tickers and Talker in a study with nine participants with VIs. Our system showed promising results as it achieved an overall mean accuracy of 93 percent in classifying activated labels. Overall, all participants with VIs reacted very positively to the system and were eager about the idea of using it. However, the evaluation results also indicated that the accuracy and bandwidth of acoustic sensing technology were limited and adding more labels would inevitably decrease the accuracy of the acoustic recognition system. In addition, acoustic sensing is sensitive to noise and may fail in noisy classrooms.

To make A3Ms more robust in noisy environment, we explored a computer vision-based sensing method. Talkit is a computer vision-based system that augments models with interactive element. To begin exploring an A3M, a user with VIs sticks a sticker on her fingernail and slides the tracker on the printed scaffold. Using an RGB camera, Talkit locates the model by sensing the tracker, and finds the user's finger by tracking the sticker. Talkit speaks audio informa-

tion when the user puts the finger with the sticker on an annotated element on the model. The user can also access more information using a speech command. We evaluated Talkit with eight participants with VIs, who managed to find five interactive elements in an A3M in 64 seconds (SD = 40) and locate a specific element in 7 seconds (SD = 8) on average.

Besides the sensing technologies, we explored the design space of A3Ms and iterated the design of our system through a series of studies. First, we explored the design space of A3Ms with a focus on input techniques. We conducted a two-section elicitation study with 12 participants with VIs. In the first section of the study, we observed participants' behavior as they explored and identified models and their components. In the second section, we elicited user-defined input techniques that would trigger various functions from an interactive model. We identified five exploration activities (e.g., comparing tactile elements), four hand postures (e.g., using one hand to hold a model in the air), and eight gestures (e.g., using index finger to strike on a model) from the participants' exploration processes and aggregate their elicited input techniques.

In addition to the design insights from people with VIs, how would teachers design models to meet their teaching plans? To tackle this question, we conducted a workshop study with Teachers of the Visually Impaired (TVIs). In the study, 35 TVIs modified sample models and added interactive elements to them during two workshops. This study provided insights about model modification (e.g., models should have salient tactile features), interactive elements (e.g., visual elements are also important to students with VIs), interaction techniques (e.g., using additional braille displays), and pedagogical considerations (e.g., models should be reusable).

With findings from the elicitation study and the workshop study, we designed a new driving application for A3Ms, Talkit++, which was tested in a co-design study with three TVIs. We worked with three TVIs individually to design A3Ms for their students. The TVIs deployed and evaluated the designed A3Ms with their students in classrooms. The findings from these studies revealed a series of design guidelines for A3Ms as teaching aids. For example, interactive models should (1) have effective tactile features (e.g., distinctive tactile patterns between components), (2) contain both auditory and visual content (e.g., explanatory animations), and (3) consider pedagogical methods (e.g., overview before details).

Lastly, we also explored two design tools for creating A3Ms: Markit and Molder. Markit is a design tool that enables designers to add interactive elements on 3D models. To add an interactive element, a designer selects a part of model in Markit and enters textual information or links to multimedia files. Markit will compile interactive elements into files that can run on Talkit and Talkit++. We evaluated Markit with eight sighted participants, who added two interactive elements to a model in 275 seconds (SD = 70) on average. Although sighted participants can use Markit to create A3Ms easily, people with VIs may have challenges using Markit because its graphical user interface relies on visual feedback.

To make the creation of A3Ms more accessible, we designed Molder. Molder is a tangible design system that enables users to modify a model and add interactive elements to the model by physically manipulating the model. Using similar algorithms from Talkit, Molder detects the position of the model and the user's behavior. When designing a model, a user needs to print a draft model,

which can be placed on a base. Then, the user can select different components of the model, and perform simple modifications like changing tactile patterns, sizes, and adding braille labels. The system also allows the user to add interactive elements. We evaluated Molder in a study with 12 participants (8 blind and low vision, 4 sighted). The participants designed tactile maps with braille labels, tactile patterns, and audio labels. The study showed that Molder was accessible and useful for map design and could be more learnable and easier to use.

Finally, we discussed the challenges and lessons we learned from these systems and studies. We discussed how to make the design and fabrication process of A3Ms easier. We also touched on new sensing technologies that may be useful for future A3Ms.

This dissertation presents an initial step towards a bigger picture. We envision that with the development of design and fabrication tools, A3Ms will be easy to design and easy to create. Everyone can design and share their A3Ms with their students, friends, and children. End users can access physical models easily and interact with them using mainstream devices. In the future, A3Ms will be new learning genre that will not only benefit people with VIs but also help other people with different needs to achieve their learning goals.

BIBLIOGRAPHY

- Abras, C., Maloney-Krichmar, D., and Preece, J. (2004). User-centered design. *Bainbridge, W. Encyclopedia of Human-Computer Interaction. Thousand Oaks: Sage Publications*, 37(4):445–456.
- Adobe (2019). Adobe Creative Cloud. Retrieved August 26, 2019 from <https://www.adobe.com/creativecloud.html>.
- AERBVI (2019). Association for Education and Rehabilitation of the Blind and Visually Impaired. Retrieved August 26, 2019 from <https://aerbvi.org/>.
- Albouys-Perrois, J., Laviolle, J., Briant, C., and Brock, A. (2018). Towards a Multisensory Augmented Reality Map for Blind and Low Vision People: a Participatory Design Approach. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*, pages 1–14. ACM.
- Apple (2019a). AVAudioPlayer - AVFoundation — Apple Developer Documentation. Retrieved August 26, 2019 from <https://developer.apple.com/documentation/avfoundation/avaudioplayer>.
- Apple (2019b). AVSpeechSynthesizer - AVFoundation — Apple Developer Documentation. Retrieved August 26, 2019 from <https://developer.apple.com/documentation/avfoundation/avspeechsynthesizer>.
- Apple (2019c). Final Cut Pro X. Retrieved August 26, 2019 from <https://www.apple.com/final-cut-pro/>.
- Apple (2019d). SFSpeechRecognizer - Speech — Apple Developer Documentation. Retrieved August 26, 2019 from <https://developer.apple.com/documentation/speech/sfspeechrecognizer>.

- Apple (2019e). UIImage - UIKit — Apple Developer Documentation. Retrieved August 26, 2019 from <https://developer.apple.com/documentation/uikit/uiimage>.
- Autodesk (2019a). Fusion 360. Retrieved August 26, 2019 from <https://www.autodesk.com/products/fusion-360/>.
- Autodesk (2019b). Maya. Retrieved August 26, 2019 from <https://www.autodesk.com/products/maya/>.
- AutoDesk (2019). Meshmixer. Retrieved August 26, 2019 from <http://www.meshmixer.com/>.
- Batshaw, M. L. (2002). *Children with disabilities*. MacLennan & Petty.
- Berg, R. E. (1995). *The Physics of Sound*. Prentice Hall.
- bld (2012). Textured Earth. Retrieved August 26, 2019 from <http://www.thingiverse.com/thing:17336>.
- Blender (2019). Free and Open 3D Creation Software. Retrieved August 26, 2019 from <https://www.blender.org/>.
- Bolt, R. A. (1980). *Put-that-there: Voice and gesture at the graphics interface*, volume 14. ACM.
- Bonnard, Q., Lemaignan, S., Zufferey, G., Mazzei, A., Cuendet, S., Li, N., Özgür, A., and Dillenbourg, P. (2013). Chilitags 2: Robust fiducial markers for augmented reality and robotics.
- Bracewell, R. N. (1986). *The Fourier transform and its applications*. McGraw-Hill.
- Bradski, G. R. and Kaehler, A. (2008). *Learning OpenCV : computer vision with the OpenCV library*. O'Reilly.

- Brock, A., Truillet, P., Oriola, B., and Jouffrais, C. (2011). Usage of multimodal maps for blind people: why and how. In *ACM International Conference on Interactive Tabletops and Surfaces (ITS '10)*, pages 247–248. ACM.
- Brock, A., Truillet, P., Oriola, B., Picard, D., and Jouffrais, C. (2012). Design and user satisfaction of interactive maps for visually impaired people. In *International Conference on Computers for Handicapped Persons*, pages 544–551. Springer.
- Brock, A. M. (2013). Touch the map!: Designing interactive maps for visually impaired people. *ACM SIGACCESS Accessibility and Computing*, (105):9–14.
- Brock, A. M., Truillet, P., Oriola, B., Picard, D., and Jouffrais, C. (2015). Interactivity Improves Usability of Geographic Maps for Visually Impaired People. *Human-Computer Interaction*, 30(2):156–194.
- Brooke, J. (1996). SUS-A quick and dirty usability scale. *Usability evaluation in industry*, 189(194):4–7.
- Brown, C. and Hurst, A. (2012). VizTouch: automatically generated tactile visualizations of coordinate spaces. In *Proceedings of the Sixth International Conference on Tangible, Embedded and Embodied Interaction (TEI '12)*, pages 131–138. ACM.
- Buehler, E., Branham, S., Ali, A., Chang, J. J., Hofmann, M. K., Hurst, A., and Kane, S. K. (2015). Sharing is Caring: Assistive Technology Designs on Thingiverse. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI '15)*, pages 525–534. ACM.
- Buehler, E., Comrie, N., Hofmann, M., McDonald, S., and Hurst, A. (2016). In-

- investigating the implications of 3d printing in special education. *ACM Transactions on Accessible Computing (TACCESS)*, 8(3):11.
- Buehler, E., Kane, S. K., and Hurst, A. (2014). ABC and 3D: opportunities and obstacles to 3D printing in special education environments. In *Proceedings of the 16th International ACM SIGACCESS Conference on Computers & Accessibility (ASSETS '14)*, pages 107–114. ACM.
- Cohen, J., Manocha, D., and Olano, M. (1997). Simplifying polygonal models using successive mappings. In *Visualization'97 (Cat. No. 97CB36155)*, pages 395–402. IEEE.
- Ducasse, J., Brock, A. M., and Jouffrais, C. (2018). Accessible Interactive Maps for Visually Impaired Users. In *Mobility of Visually Impaired People*, pages 537–584. Springer.
- Dufaux, A. (2001). *Detection and Recognition of Impulsive Sound Signals*. PhD thesis, Verlag nicht ermittelbar.
- Espinosa, M. A., Ungar, S., Ocharita, E., Blades, M., and Spencer, C. (1998). Comparing methods for introducing blind and visually impaired people to unfamiliar urban environments. *Journal of Environmental Psychology*, 18(3):277–287.
- Geißler, J. (1995). Gedrics: The Next Generation of Icons. In *HumanComputer Interaction*, pages 73–78. Springer.
- Giraud, S., Brock, A. M., Macé, M. J.-M., Jouffrais, C., Chetouani, M., Fyfe Pietro Gillies, M., Juan, M.-C., Jouffrais, C., Giraud, S., Brock, A. M., and J-M Macé, M. (2017). Map Learning with a 3D Printed Interactive Small-Scale Model: Improvement of Space and Text Memorization in Visually Impaired Students. *Frontiers in Psychology*, 8:930.

- Good, M. D., Whiteside, J. A., Wixon, D. R., and Jones, S. J. (1984). Building a user-derived interface. *Communications of the ACM*, 27(10):1032–1043.
- Götzelmann, T. (2016). LucentMaps: 3D Printed Audiovisual Tactile Maps for Blind and Visually Impaired People. In *Proceedings of the 18th International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS '16)*, pages 81–90. ACM.
- Götzelmann, T. (2018). Visually augmented audio-tactile graphics for visually impaired people. *ACM Transactions on Accessible Computing (TACCESS)*, 11(2):8.
- Götzelmann, T. and Pavkovic, A. (2014). Towards automatically generated tactile detail maps by 3d printers for blind persons. In *International Conference on Computers for Handicapped Persons*, pages 1–7. Springer.
- GrabCAD (2019). Design Community, CAD Library, 3D Printing Software. Retrieved August 26, 2019 from <https://grabcad.com/>.
- Grice, N., Christian, C., Nota, A., and Greenfield, P. (2015). 3D Printing Technology: A Unique Way of Making Hubble Space Telescope Images Accessible to Non-Visual Learners. *Journal of Blindness Innovation and Research*, 5(1).
- Gual, J., Puyuelo, M., and Lloveras, J. (2011). Universal design and visual impairment: Tactile products for heritage access. In *Proceedings of the 18th International Conference on Engineering Design (ICED 11), Impacting Society Through Engineering Design*, volume 5, pages 155–164.
- Gual, J., Puyuelo, M., and Lloveras, J. (2014). Three-dimensional tactile symbols produced by 3D Printing: Improving the process of memorizing a tactile map key. *British Journal of Visual Impairment*, 32(3):263–278.

- Gual, J., Puyuelo, M., Lloveras, J., and Merino, L. (2012). Visual Impairment and urban orientation. Pilot study with tactile maps produced through 3D Printing. *Psychology*, 3(2):239–250.
- Guiard, Y. (1987). Asymmetric Division of Labor in Human Skilled Bimanual Action. *Journal of Motor Behavior*, 19(4):486–517.
- Harrison, C., Tan, D., and Morris, D. (2010). Skinput: appropriating the body as an input surface. In *Proceedings of the 28th international conference on Human factors in computing systems (CHI '10)*, pages 453–462. ACM.
- Hatch, M. (2013). *The Maker Movement Manifesto: Rules for Innovation in the New World of Crafters, Hackers, and Tinkerers*. McGraw Hill Professional.
- Hatwell, Y. (1987). Motor and cognitive functions of the hand in infancy and childhood. *International Journal of Behavioral Development*, 10(4):509–526.
- Hatwell, Y., Arlette, S., and Edouard, G. (2003). *Touching for knowing: cognitive psychology of haptic manual perception*, volume 53. John Benjamins Publishing.
- Hinterstoisser, S., Lepetit, V., Ilic, S., Holzer, S., Bradski, G., Konolige, K., and Navab, N. (2012). Model Based Training, Detection and Pose Estimation of Texture-Less 3D Objects in Heavily Cluttered Scenes. In *Asian conference on computer vision*, pages 548–562. Springer.
- Holloway, L., Marriott, K., and Butler, M. (2018). Accessible Maps for the Blind : Comparing 3D Printed Models with Tactile Graphics. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*, pages Paper 198, 13 pages. ACM.
- Hook, J., Nappey, T., Hodges, S., Wright, P., and Olivier, P. (2014). Making 3D printed objects interactive using wireless accelerometers. In *Proceedings*

- of the extended abstracts of the 32nd annual ACM conference on Human factors in computing systems (CHI EA '14)*, pages 1435–1440. ACM.
- Hu, M. (2015). Exploring New Paradigms for Accessible 3D Printed Graphs. In *Proceedings of the 17th International ACM SIGACCESS Conference on Computers & Accessibility (ASSETS '15)*, pages 365–366. ACM.
- Huggins-Daines, D., Kumar, M., Chan, A., Black, A. W., Ravishankar, M., and Rudnicky, A. I. (2006). Pocketsphinx: A free, real-time continuous speech recognition system for hand-held devices. In *2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings*, volume 1, pages 185–188. IEEE.
- Huo, K., Vinayak, and Ramani, K. (2017). Window-Shaping: 3D Design Ideation by Creating on, Borrowing from, and Looking at the Physical World. In *Proceedings of the Tenth International Conference on Tangible, Embedded, and Embodied Interaction (TEI '17)*, pages 37–45. ACM.
- Hurst, A. and Tobias, J. (2011). Empowering individuals with do-it-yourself assistive technology. In *The proceedings of the 13th international ACM SIGACCESS conference on Computers and accessibility (ASSETS '11)*, pages 11–18. ACM.
- Ishiguro, Y. and Poupyrev, I. (2014). 3D printed interactive speakers. *Proceedings of the 32nd annual ACM conference on Human factors in computing systems (CHI '14)*, pages 1733–1742.
- Kane, S. K. and Bigham, J. P. (2014). Tracking@ stemxcomet: teaching programming to blind students via 3D printing, crisis management, and twitter. In *Proceedings of the 45th ACM technical symposium on Computer science education (SIGCSE '14)*, pages 247–252. ACM.

- Kim, J. and Yeh, T. (2015). Toward 3D-Printed Movable Tactile Pictures for Children with Visual Impairments. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI '15)*, pages 2815–2824. ACM.
- Kitchin, R. and Freundschuh, S. (2000). *Cognitive mapping : past, present, and future*. Routledge.
- Kolitsky, M. A. (2014). 3D Printed Tactile Learning Objects: Proof of Concept. *Journal of Blindness Innovation and Research*, 4(1):4–51.
- kristinaprofitt1 (2016). Mt. Justimore - Thingiverse.
- Kumar, D., Ramasamy, R., and Stefanich, G. P. (2001). Science for students with visual impairments: Teaching suggestions and policy implications for secondary educators. *Electronic journal of science education*, 5(3).
- Laput, G., Ahuja, K., Goel, M., and Harrison, C. (2018). Ubioustics: Plug-and-Play Acoustic Activity Recognition Ubioustics: Plug-and-Play Acoustic Activity Recognition. In *The 31st Annual ACM Symposium on User Interface Software and Technology (UIST '18)*, pages 213–224. ACM.
- Laput, G., Brockmeyer, E., Hudson, S. E., and Harrison, C. (2015). Acoustruments: Passive, Acoustically-Driven, Interactive Controls for Handheld Devices. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI '15)*, pages 2161–2170. ACM.
- Lederman, S. J. and Klatzky, R. L. (1987). Hand movements: A window into haptic object recognition. *Cognitive psychology*, 19(3):342–368.
- Lederman, S. J. and Klatzky, R. L. (1993). Extracting object properties through haptic exploration. *Acta psychologica*, 84(1):29–40.

- Lilley, I. M. (1967). *Friedrich Froebel: A selection from his writings*. Cambridge University Press.
- Lindstrom, P. (2000). Out-of-core simplification of large polygonal models. In *Proceedings of the 27th annual conference on Computer graphics and interactive techniques (SIGGRAPH '00)*, pages 259–262. ACM Press/ Addison-Wesley Publishing Co.
- Lipson, H. and Kurman, M. (2013). *Fabricated: The new world of 3D printing*. John Wiley & Sons.
- Lobben, A. (2015). Tactile maps and mapping. *Journal of Blindness Innovation and Research*, 5(1).
- louisheath (2017). braille-printer: A python program which produces 3D printable braille signs from plain text. CSS Bloomberg Hackathon 2017. Retrieved August 26, 2019 from <https://github.com/louisheath/braille-printer>.
- Makerbot (2019). 3D Printers for Educators, Professionals — MakerBot. Retrieved August 26, 2019 from <https://www.makerbot.com/>.
- McDonald, S., Dutterer, J., Abdolrahmani, A., Kane, S. K., and Hurst, A. (2014). Tactile aids for visually impaired graphical design education. In *Proceedings of the 16th International ACM SIGACCESS Conference on Computers & Accessibility (ASSETS '14)*, pages 275–276. ACM.
- Merrill, M. D., Drake, L., Lacy, M. J., and Pratt, J. (1996). Reclaiming instructional design. *Educational Technology*, 36(5):5–7.
- mfrtiz (2016). 3D Print a Volcano by mfrtiz - Thingiverse. Retrieved August 26, 2019 from <https://www.thingiverse.com/thing:1316498>.

- Microsoft PowerPoint (2019). Slide Presentation Software. Retrieved August 26, 2019 from <https://products.office.com/en-us/powerpoint>.
- Morash, V. S., Pensky, A. E. C., Tseng, S. T., and Miele, J. A. (2014). Effects of using multiple hands and fingers on haptic performance in individuals who are blind. *Perception*, 43(6):569–588.
- Morris, M. R. (2012). Web on the Wall: Insights from a Multimodal Interaction Elicitation Study. In *Proceedings of the 2012 ACM international conference on Interactive tabletops and surfaces (ITS '12)*, pages 95–104.
- Morris, M. R., Danielescu, A., Drucker, S., Fisher, D., Lee, B., and Wobbrock, J. O. (2014). Reducing legacy bias in gesture elicitation studies. *interactions*, 21(3):40–45.
- Mueller, S., Im, S., Gurevich, S., Teibrich, A., Pfisterer, L., Guimbretière, F., and Baudisch, P. (2014). Wireprint: 3d printed previews for fast prototyping. In *Proceedings of the 27th annual ACM symposium on User interface software and technology (UIST '14)*, pages 273–280. ACM.
- Murray-Smith, R., Williamson, J., Hughes, S., and Quaade, T. (2008). Stane: synthesized surfaces for tactile input. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '08)*, pages 1299–1302. ACM.
- Nakagaki, K., Fitzgerald, D., Ma, Z. J., Vink, L., Levine, D., and Ishii, H. (2019). inFORCE: Bi-directional ‘Force’ Shape Display for Haptic Interaction. In *Proceedings of the Thirteenth International Conference on Tangible, Embedded, and Embodied Interaction (TEI '19)*, pages 615–623. ACM.
- National Federation of the Blind, Jernigan Institute (2009). The Braille Literacy Crisis in America. *National Federation of the Blind*.

- Nielsen, J. (1994). Guerrilla HCI: Using discount usability engineering to penetrate the intimidation barrier. *Cost-justifying usability*, pages 245–272.
- Norman, D. A. (1988). *The psychology of everyday things*. Basic books.
- Norman, D. A., Draper, S. W., and Taylor & Francis. (1986). *User centered system design : new perspectives on human-computer interaction*. CRC Press.
- OpenSCAD (2019). The Programmers Solid 3D CAD Modeller. Retrieved August 26, 2019 from <https://www.openscad.org/>.
- OpenStreetMap (2019). OpenStreetMap. Retrieved August 26, 2019 from <https://www.openstreetmap.org/>.
- OSM Buildings (2019). OSM Buildings. Retrieved August 26, 2019 from <https://osmbuildings.org/>.
- PaLEoS (2015). Animal Cell. Retrieved August 26, 2019 from <http://www.thingiverse.com/thing:689381>.
- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al. (2011). Scikit-learn: Machine learning in python. *Journal of machine learning research*, 12(Oct):2825–2830.
- Peng, H., Briggs, J., Wang, C.-Y., Guo, K., Kider, J., Mueller, S., Baudisch, P., and Guimbretière, F. (2018). RoMA: Interactive Fabrication with Augmented Reality and a Robotic 3D Printer. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*. ACM.
- Peng, H., Guimbretière, F., Mccann, J., and Hudson, S. E. (2016). A 3D Printer for

- Interactive Electromagnetic Devices. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology (UIST '16)*, pages 553–562. ACM.
- Peng, H., Mankoff, J., Hudson, S. E., and McCann, J. (2015). A Layered Fabric 3D Printer for Soft Interactive Objects. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI'15)*, pages 1789–1798. ACM.
- Pestalozzi, J. H. (1803). *ABC der Anschauung, oder, Anschauungs-Lehre der Massverhältnisse*, volume 2. Geßner.
- Piskurich, G. M. (2015). *Rapid instructional design: Learning ID fast and right*. Wiley.
- Qian, C., Sun, X., Wei, Y., Tang, X., and Sun, J. (2014). Realtime and robust hand tracking from depth. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1106–1113.
- Reichinger, A., Fuhrmann, A., Maierhofer, S., and Purgathofer, W. (2016). Gesture-Based Interactive Audio Guide on Tactile Reliefs. In *Proceedings of the 18th International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS '16)*, pages 91–100. ACM.
- Reiser, R. A. (2001). A history of instructional design and technology. *Educational technology research and development*, 49(1):53.
- Rhino (2019). Rhino 6 for Windows and Mac. Retrieved August 26, 2019 from <https://www.rhino3d.com/>.
- Roederer, J. G. (2008). *The Physics and Psychophysics of Music: An Introduction*. Springer Science & Business Media.

- Rossing, T. D. (2000). *Science of Percussion Instruments*. World Scientific.
- Rowell, J. and Ungar, S. (2003). The world of touch: an international survey of tactile maps. Part 1: production. *British Journal of Visual Impairment*, 21(3):98–104.
- Rule, A. C., Stefanich, G. P., Boody, R. M., and Peiffer, B. (2011). Impact of adaptive materials on teachers and their students with visual impairments in secondary science and mathematics classes. *International Journal of Science Education*, 33(6):865–887.
- Saldana, J. (2015). *The Coding Manual for Qualitative Researchers*. Sage.
- Sato, M., Poupyrev, I., and Harrison, C. (2012). Touché: enhancing touch interaction on humans, screens, liquids, and everyday objects. In *Proceedings of the 2012 ACM annual conference on Human Factors in Computing Systems (CHI '12)*, pages 483–492. ACM.
- Savage, V., Chang, C., and Hartmann, B. (2013). Sauron: embedded single-camera sensing of printed physical user interfaces. In *Proceedings of the 26th annual ACM symposium on User interface software and technology (UIST '13)*, pages 447–456. ACM.
- Savage, V., Follmer, S., Li, J., and Hartmann, B. (2015a). Makers' Marks: Physical Markup for Designing and Fabricating Functional Objects. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology - UIST '15*, pages 103–108, New York, New York, USA. ACM Press.
- Savage, V., Head, A., Hartmann, B., Goldman, D. B., Mysore, G., and Li, W. (2015b). Lamello: Passive Acoustic Sensing for Tangible Input Components.

- In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI '15)*, pages 1277–1280. ACM.
- Savage, V., Schmidt, R., Grossman, T., Fitzmaurice, G., and Hartmann, B. (2014). A series of tubes: adding interactivity to 3D prints using internal pipes. In *Proceedings of the 27th annual ACM symposium on User interface software and technology (UIST '14)*, pages 3–12. ACM.
- Schmitz, M., Khalilbeigi, M., Balwierz, M., Lissermann, R., Mühlhäuser, M., and Steimle, J. (2015). Capricate: A Fabrication Pipeline to Design and 3D Print Capacitive Touch Sensors for Interactive Objects. *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology (UIST '15)*, pages 253–258.
- Scouten, E. L. (1967). The Rochester method, an oral multisensory approach for instructing prelingual deaf children. *American annals of the deaf*, 112(2):50–5.
- Sharp, T., Keskin, C., Robertson, D., Taylor, J., Shotton, J., Kim, D., Rhemann, C., Leichter, I., Vinnikov, A., and Wei, Y. (2015). Accurate, robust, and flexible real-time hand tracking. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI '15)*, pages 3633–3642. ACM.
- Shen, H., Edwards, O., Miele, J., and Coughlan, J. M. (2013). CamIO: a 3D computer vision system enabling audio/haptic interaction with physical objects by blind users. In *Proceedings of the 15th International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS '13)*, pages 1–2. ACM.
- Sheppard, L. and Aldrich, F. K. (2001). Tactile graphics in school education: perspectives from teachers. *British Journal of Visual Impairment*, 19(3):93–97.

- Shi, L. (2015). Talkabel: A Labeling Method for 3D Printed Models. In *Proceedings of the 17th International ACM SIGACCESS Conference on Computers & Accessibility (ASSETS '15)*, pages 361–362. ACM.
- Shi, L., Lawson, H., Zhang, Z., and Azenkot, S. (2019). Designing Interactive 3D Printed Models with Teachers of the Visually Impaired. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems (CHI '19)*, pages 197:1–197:14. ACM.
- Shi, L., McLachlan, R., Zhao, Y., and Azenkot, S. (2016a). Magic Touch: Interacting with 3D Printed Graphics. In *Proceedings of the 18th International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS '16)*, pages 329–330. ACM.
- Shi, L., Zelzer, I., Feng, C., and Azenkot, S. (2016b). Tickers and Talker: An Accessible Labeling Toolkit for 3D Printed Models. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*, pages 4896–4907. ACM.
- Shi, L., Zhang, Z., and Azenkot, S. (2018). A Demo of Talkit++. In *Proceedings of the 20th International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS '18)*, pages 429–431. ACM.
- Shi, L., Zhao, Y., and Azenkot, S. (2017a). Designing Interactions for 3D Printed Models with Blind People. In *In Proceedings of the 19th International ACM SIGACCESS Conference on Computers & Accessibility (ASSETS '17)*, pages 200–209. ACM.
- Shi, L., Zhao, Y., and Azenkot, S. (2017b). Markit and Talkit: A Low-Barrier Toolkit to Augment 3D Printed Models with Audio Annotations. In *Proceed-*

- ings of the 30th Annual ACM Symposium on User Interface Software and Technology (UIST '17), pages 493–506. ACM.
- Shneiderman, B., Plaisant, C., Cohen, M. M. S., Jacobs, S. M., and Elmqvist, N. (2016). *Designing the user interface : strategies for effective human-computer interaction*. Pearson.
- SketchUp (2019). 3D Design Software — 3D Modeling on the Web — SketchUp. Retrieved August 26, 2019 from <https://www.sketchup.com/>.
- SolidWorks (2019). 3D CAD Design Software. Retrieved August 26, 2019 from <https://www.solidworks.com/>.
- Song, H., Guimbretière, F., Hu, C., and Lipson, H. (2006). ModelCraft: capturing freehand annotations and edits on physical 3D models. In *Proceedings of the 19th annual ACM symposium on User interface software and technology (UIST '06)*, pages 13–22. ACM.
- Stangl, A., Hsu, C.-L., and Yeh, T. (2015). Transcribing across the senses: community efforts to create 3D printable accessible tactile pictures for young children with visual impairments. In *Proceedings of the 17th International ACM SIGACCESS Conference on Computers & Accessibility (ASSETS '15)*, pages 127–137. ACM.
- Stangl, A., Kim, J., and Yeh, T. (2014). 3D printed tactile picture books for children with visual impairments. In *Proceedings of the 2014 conference on Interaction design and children (IDC '14)*, pages 321–324. ACM.
- Swaminathan, S., Roumen, T., Kovacs, R., Stangl, D., Mueller, S., and Baudisch, P. (2016). Linespace: A sensemaking platform for the blind. In *Proceedings*

- of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16), pages 2175–2185. ACM.
- Szpiro, S., Zhao, Y., and Azenkot, S. (2016). Finding a store, searching for a product: a study of daily challenges of low vision people. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp '16)*, pages 61–72. ACM.
- Taylor, B., Dey, A., Siewiorek, D., and Smailagic, A. (2016). Customizable 3D Printed Tactile Maps as Interactive Overlays. In *Proceedings of the 18th International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS '16)*, pages 71–79. ACM.
- Taylor, B. T., Dey, A. K., Siewiorek, D. P., and Smailagic, A. (2015). TactileMaps.net: A web interface for generating customized 3D-printable tactile maps. In *Proceedings of the 17th International ACM SIGACCESS Conference on Computers & Accessibility (ASSETS '15)*, pages 427–428. ACM.
- Tejada, C., Li, Z., and Ashbrook, D. (2018). Blowhole : Blowing-Activated Tags for Interactive 3D-Printed Models. In *Proceedings of the 44th Graphics Interface Conference*, pages 122–128.
- Teske, J., Gray, P., Clausen, C., Smith, L. L., Alsubia, S., Ghayoorad, M., Rule, A. C., and Schneider, J. (2014). Teacher-Made Tactile Science Materials with Critical and Creative Thinking Activities for Learners Including Those with Visual Impairments.
- Thingiverse (2019). Thingiverse - Digital Designs for Physical Objects. Retrieved August 26, 2019 from <https://www.thingiverse.com/>.

- Thinkable (2019). TactileView - Look and feel, feel and look. Retrieved August 26, 2019 from <http://www.tactileview.com/>.
- Tinkercad (2019). Tinkercad — Create 3D digital designs with online CAD. Retrieved August 26, 2019 from <https://www.tinkercad.com/>.
- Tiptoi® (2019). Discover the world of Tiptoi®. Retrieved August 26, 2019 from <https://www.ravensburger.de/entdecken/ravensburger-marken/tiptoi/index.html>.
- Touch Mapper (2019). Tactile Maps Easily. Retrieved August 26, 2019 from <https://touch-mapper.org/en/>.
- Tzanetakis, G. and Cook, P. (2002). Musical genre classification of audio signals. *IEEE Transactions on Speech and Audio Processing*, 10(5):293–302.
- Ultimaker (2019). Ultimaker PVA material: Water-soluble support for complex prints. Retrieved August 26, 2019 from <https://ultimaker.com/materials/pva>.
- Vázquez, M., Brockmeyer, E., Desai, R., Harrison, C., and Hudson, S. E. (2015). 3D Printing Pneumatic Device Controls with Variable Activation Force Capabilities. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI '15)*, pages 1295–1304. ACM.
- vvoovv (2019). blender-osm: One click download and import of OpenStreetMap and terrain for Blender! Global coverage! Documentation is hosted here. Retrieved August 26, 2019 from <https://github.com/vvoovv/blender-osm>.
- Wang, J., Zhao, K., Zhang, X., and Peng, C. (2014). Ubiquitous keyboard for

- small mobile devices. In *Proceedings of the 12th annual international conference on Mobile systems, applications, and services (MobiSys '14)*, pages 14–27. ACM.
- WaveSkyLord (2016). Slice View Volcano - Thingiverse. Retrieved August 26, 2019 from <https://www.thingiverse.com/thing:1290606>.
- Wedler, H. B., Cohen, S. R., Davis, R. L., Harrison, J. G., Siebert, M. R., Willenbring, D., Hamann, C. S., Shaw, J. T., and Tantillo, D. J. (2012). Applied Computational Chemistry for the Blind and Visually Impaired. *Journal of Chemical Education*, 89(11):1400–1404.
- Weichel, C., Lau, M., Kim, D., Villar, N., Gellersen, H. W., Weichel, C., Lau, M., Kim, D., Villar, N., and Gellersen, H. W. (2014). MixFab: a mixed-reality environment for personal fabrication. In *Proceedings of the 32nd annual ACM conference on Human factors in computing systems (CHI '14)*, pages 3855–3864. ACM.
- Willis, K., Brockmeyer, E., Hudson, S., and Poupyrev, I. (2012). Printed optics: 3D printing of embedded optical elements for interactive devices. In *Proceedings of the 25th annual ACM symposium on User interface software and technology (UIST '12)*, pages 589–598. ACM.
- Wobbrock, J. O., Morris, M. R., and Wilson, A. D. (2009). User-defined gestures for surface computing. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '09)*, pages 1083–1092.
- xehpuk (2013). Simplified Tiger Tank. Retrieved August 26, 2019 from <http://www.thingiverse.com/thing:90265>.
- YahooJAPAN (2013). Plane by YahooJAPAN - Thingiverse. Retrieved August 26, 2019 from <https://www.thingiverse.com/thing:182252>.

- Yeh, T. and Kim, J. (2018). CraftML: 3D Modeling is Web Programming. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems (CHI '18)*, pages 527:1–527:12. ACM.
- Zhang, H. and Hoff, K. E. (1997). Fast backface culling using normal masks. In *Proceedings of the 1997 symposium on Interactive 3D graphics (SI3D '97)*, pages 103–106. ACM.
- Zhang, Y., Laput, G., and Harrison, C. (2017). Electrick : Low-cost touch sensing using electric field tomography. In *In Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17)*, pages 1–14. ACM.
- Zuckerman, O., Arida, S., and Resnick, M. (2005). Extending tangible interfaces for education: digital montessori-inspired manipulatives. In *Proceedings of the SIGCHI conference on Human factors in computing systems (CHI '05)*, pages 859–868. ACM.