Markit and Talkit: A Low-Barrier Toolkit to Augment 3D Printed Models with Audio Annotations

Lei Shi
ls776@cornell.edu

Yuhang Zhao
yz769@cornell.edu

Shiri Azenkot
shiri.azenkot@cornell.edu

Jacobs Technion-Cornell Institute, Cornell Tech
New York, NY, USA

ABSTRACT

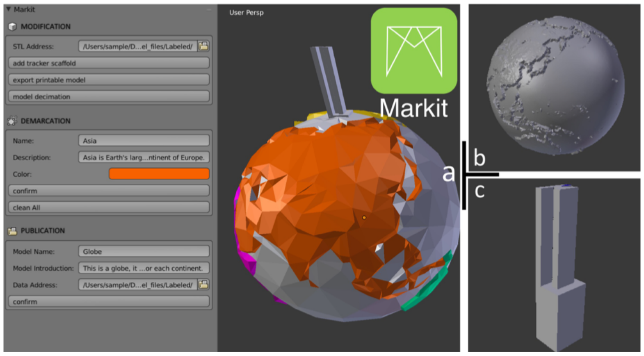
As three-dimensional printers become more available, 3D printed models can serve as important learning materials, especially for blind people who perceive the models tactilely. Such models can be much more powerful when augmented with *audio annotations* that describe the model and their elements. We present Markit and Talkit, a low-barrier toolkit for creating and interacting with 3D models with audio annotations. Makers (e.g., hobbyists, teachers, and friends of blind people) can use Markit to mark model elements and associate then with text annotations. A blind user 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 Markit with eight sighted "makers" and Talkit with eight blind people. On average, non-experts added two annotations to a model in 275 seconds (SD=70) with Markit. Meanwhile, with Talkit, blind people found a specified annotation on a model in an average of 7 seconds (SD=8).

Author Keywords
Visual impairments; 3D models; computer vision

ACM Classification Keywords
H.5.2. Information Interfaces and Presentation: User Interfaces - Input devices and strategies.

INTRODUCTION

Three-dimensional models are valuable learning tools, especially for blind people who cannot access 2D visualizations. Access to 3D models has a major impact on blind people's education and daily lives. For example, when studying abstract concepts in chemistry and astronomy, teachers advocate that blind students should use 3D models to learn the structure of molecules and celestial bodies [42,45]. Studies have shown that blind students who studied with models performed better on exams [46] and engaged more in activities together with their sighted peers [63].



STL Address: /Users/sample/D...el_files/Labeled/
Name: Asia
Description: Asia is Earth's larg...ntinent of Europe.
Color:
Model Name: Globe
Model Introduction: This is a globe, it ...or each continent.
Data Address: /Users/sample/D...el_files/Labeled/

Figure 1. A maker downloads a model of a globe (a), and imports it into Markit. She adds the tracker scaffold (c) and associates text with areas on the model (b).

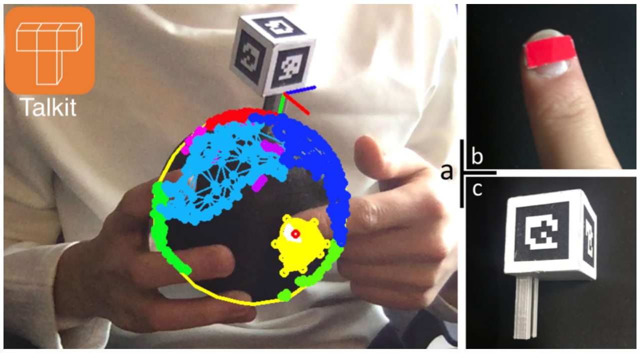


Figure 2. A blind user prints the modified globe shown in Figure 1. She sticks a simple sticker on her fingernail (b), and slides the tracker on the printed scaffold (c). She then launches the Talkit application (a) on her computer and interactively explores the model with its audio annotations.

Since 3D models are such important learning tools for blind students, 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 blind students [2,3]. 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 [46,63]. 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.

UIST 2017, October 22–25, 2017, Quebec City, QC, Canada

ACM 978-1-4503-4981-9/17/10…$15.00
https://doi.org/10.1145/3126594.3126650

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 [16], but they can also enable blind students to generate their own tactile visualizations [27]. The maker movement [22] has spawned a global community of hobbyists, who are eager to share resources and solve socially-motivated problems; thousands, if not millions of "makers," around the world create and share 3D models and accessibility solutions [13,33,34,64]. For example, makers, who are hobbyists, amateurs, and friends of blind people, shared models designed to help blind people learn concepts on Thingiverse [13].

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 for blind people [26,44], only a small minority of people with visual impairments read braille [37]. Moreover, braille requires a lot of space and can easily be confused with other tactile features in a model [55]. Thus, blind people, especially those who don't know braille, can have trouble understanding these models, and must rely on additional assistance from other people.

Facing these opportunities and challenges, we designed *Markit* and *Talkit*, components of a toolkit that augments 3D models with audio annotations. Markit empowers makers (hobbyists of 3D printing and similar technologies) to create augmented models with annotations in several minutes. Talkit is an application, which enables blind users to interactively explore the annotations on the models. The toolkit only needs a commodity camera and a microphone and doesn't require complicated assembly, additional electronics, or special materials.

Markit is an add-on to Blender [7], a popular 3D modeling application among designers. It provides step-by-step guidance for creating augmented models for non-experts, as shown in Figure 1. After importing a downloaded model into Markit, a maker can add the tracker scaffold (Figure 1c) and associate textual annotations with elements on the model (Figure 1b).

Talkit is an application for a personal computer that allows a blind user to interact with an augmented model (Figure 2). To begin exploring a printed model, a blind user sticks a sticker on her fingernail (Figure 2b), and slides the tracker (Figure 2c) on the printed scaffold. Using a 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 Markit and Talkit in two studies, with sighted and blind participants, respectively. Our results showed that sighted participants learnt how to use Markit quickly and added two annotations to a model in an average of 275 seconds (SD=70). When using Talkit, blind participants 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 summary, we contribute:

1. Markit, a Blender add-on that enables non-experts to easily and efficiently create augmented models with annotations.
2. Talkit, an application that allows blind users to interact with augmented models and access audio annotations using gestures and speech commands.

RELATED WORK

3D Printing Assistive Technologies

The growing availability of 3D printers enables people to create customizable assistive technologies [13,14,25]. Notably, people have created specialized devices for prosthetics users [23] and people with motor and mobility impairments [15].

For blind people, 3D printed models and graphics are promising tools for learning new concepts and performing daily activities. Researchers proposed to use 3D printed visualizations in curriculums like programming [27], design [35], and mathematics [11,16]. Visually impaired children can learn basic literacy skills from printed tactile books [29,59]. In addition to educational applications, 3D printing can also make daily tasks easier. For example, Facade [20] enables blind people to independently add 3D printed tactile buttons to appliances. Linespace [61] is a printing system that allows blind people to understand daily visualizations such as maps.

Our work leverages the growing prevalence of 3D printers and makes 3D printed models more powerful educational tools for blind as well as sighted people.

Interactive 3D Prints

In recent decades, researchers contributed different techniques to create interactive printed objects for various uses. Most of these prior techniques involved conductive materials, embedded electronic components, or special acoustic structures. Unlike prior work, Talkit and Markit used commodity cameras and computer vision technology.

Researchers have used conductive materials to add interactivity to 3D models. Capricate [51] is a tool that allows users to design 3D prints with embedded capacitive materials. Similarly, Götzelmann [18], Kolisky [31], Brule *et al*. [12], and Taylor *et al*. [62] used conductive materials and touchscreens to create interactive 3D printed models. When a user touched a conductive part of a model, the touchscreen could recognize the touch gesture and play an audio label. However, the conductive part of the model has to be in contact with a touchscreen, so a user cannot explore the model freely in the air. In addition to touchscreens, makers and researchers also used Arduino controllers [30] and electric field tomography [69] to sense a user's input on conductive 3D printed objects.

A variety of projects embedded electronic components in 3D printed objects to create interactive applications. Researchers proposed to create functional 3D printed prototypes by embedding a camera in a 3D print [47], electromechanical components [48], electronics combined with pipes [50,66], and pneumatic sensors [65]. These techniques enable designers to fabricate functional early prototypes for testing. In addition, researchers developed printers that print electronic components inside 3D objects [40,41]. While these methods contributed to different sensing techniques and interactions with 3D printed objects, they also involved specific design skills and expenses.

Acoustic sensing has been used to make 3D prints interactive without additional electronics or specialized materials. For example, Shi *et al.* [53,55] used passive acoustic sensing to enable 3D printed models to provide audio feedback. They designed models with printable percussive components, which have special structures and unique acoustic features. An acoustic sensing application recognizes the sound generated from the percussive components, and speaks the associated labels. However, 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. Similarly, Lamello [49] and Stane [36] are tangible controllers that detect user input on 3D prints using special structures and passive acoustic sensing. Besides passive acoustic sensing, active acoustic sensing like Acoustruments [32], which combined the speaker and microphone of a smartphone together as an ultrasonic scanner, can create tangible applications for handheld devices (e.g., transforming a mobile phone into an interactive toy car).

Compared to prior techniques, Talkit and Markit leverage computer vision technology, without requiring complicated assembly, embedded electronics, specialized materials, or special structures. An amateur can easily add audio annotations to a model with minimal modifications, and a blind user can interact with the original model without the constraints of embedded electronics and touchscreens.

Accessing Annotations Using Computer Vision

Computer vision has been applied to augment 3D objects and create interactive applications. Besides creating interactive printed models with professional camera systems [1] and annotating 3D objects using a printed tracking pattern [58], some researchers also developed systems that use commodity cameras to help blind people access annotations on 2D and 3D prints in their daily lives.

Previous work used different types of cameras to make 2D printed materials and labels more accessible to blind people. For example, blind people could use finger-worn cameras to read printed text [57,60]. In addition, researchers used desk-mounted cameras, head-mounted cameras, smartphones' and tablets' cameras to access text and annotations on printed materials [28], appliances [19], and tactile graphics [4,17].

While most work targeted text and labels on 2D prints, there were only a few projects involving 3D objects. These projects used mounted depth cameras to annotate 3D objects. CamIO [52] used a Kinect camera to identify objects with fiducial markers. Similarly, Reichinger et al. [43] used the Intel RealSense F200 as a depth sensor and developed a system that allowed blind people to explore tactile reliefs. However, these projects required a blind user to place models in a fixed position on a table to retrieve information, which prevented the user from freely exploring the models.

Unlike prior work, Talkit allows blind users to fully explore the 3D spatial information on models with a single RGB camera on a mobile device. With makers labeling the 3D models using Markit, a blind user can potentially learn from any online models.

DESIGN EXPLORATION

In this section, we describe our design goals and an exploratory study we conducted with blind people to guide the design of Talkit.

Design Goals

In this paper, we focus on enabling people to annotate 3D prints with audio annotations. Currently, to add interactive annotations, makers usually need to design a model from a sketch, assemble electronics and wires, and sometimes hardcode annotations in software. These tasks are time-consuming and require skills that a lot of hobbyists don't have. In addition, most makers are not specialized in assistive technologies and their models may not fulfill blind people's needs. Thus, while possible, there are no existing tools that enable people to easily add annotations to models that can later be accessed effectively by blind people.

To address these barriers, we formulated design goals that consider the characteristics of our two target user groups: makers and blind people. These design goals are:

Create a simple tool for makers to annotate models. We aim to design an easy-to-use tool that enables non-expert makers to add annotations to 3D models.

Provide blind users easy access to a model's annotations. The audio augmented models should not require complex assembly or expensive auxiliary components. After printing a model, a blind person should be able to access its annotations with little added effort.

Interacting with an augmented 3D model should not inhibit a blind user's ability to naturally explore the model. Blind people should be able to tactilely explore a model and access the individual annotations efficiently. In prior work [55], we found that blind people sometimes wanted to hold a model with both hands, move it around, and rotate it to learn about its structure and texture. Previous projects that required a 3D print to be fixed to a surface or attached to a device inhibited a user's ability to explore the model [12,18,31,43,52,62].

Exploratory Study

We conducted an exploratory study to develop effective interaction techniques for accessing annotations from tactile models. We aim to understand what interactions do blind people prefer to use when exploring interactive 3D printed models. For the full details and analysis of the study, see Shi *et al*. [56]. We recruited 12 blind participants (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 augmented models and asked each participant to design different gestures and interaction techniques with three printed models: a globe, a cell, and a map model. We asked them to explore the models 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."

In addition to finger gestures, participants mentioned other input methods like speech commands and buttons. Five participants wanted to use simple commands such as "what is it?" to access annotations. Three participants mentioned buttons. For example, P3 designed buttons to get detailed information for a continent on the Globe model.

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." Holding a model of a cell, he added, "It's annoying, [if the system would say] 'nucleus,' 'nucleus,' 'nucleus,' every time you touch it."

TOOLKIT DESIGN

We designed Markit and Talkit based on our design goals and findings from the exploratory study. 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're more available than depth cameras in commodity devices blind people are using today (e.g., iPhones). 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 RBG cameras, we may face some tradeoffs. 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.

Markit

Markit is a design tool for makers to create augmented models from online resources.

Interaction Flow

As shown in Figure 3, a user follows three steps to label an existing model, which can be downloaded from online resources:

1. 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.
2. Demarcation. The user marks an element with a specific color and annotate it with a name and detailed description.
3. Publication. The user uses Markit to export the model's annotation data, which can be read by Talkit.

Implementation

Markit is an add-on for Blender, 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.

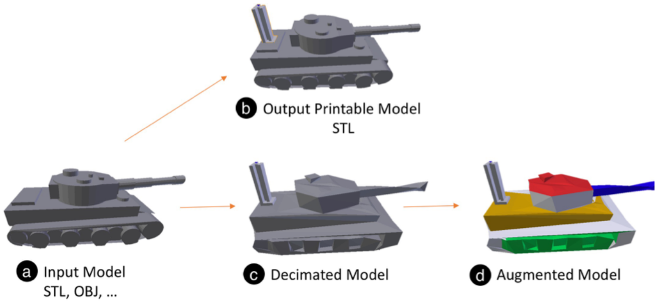


Figure 3. For an input model (a), a user uses Markit to add a tracker scaffold. Markit outputs a printable model in STL format (b) and decimates the model (c) for future edits. Then, the user marks the model in colors and adds annotations on it (d). Finally, Markit publishes the model's annotation data.

The User Interface

As shown in Figure 4, Markit has three sections in its toolbar on the user interface: Modification, Demarcation, and Publication. In Blender, a model is a triangle mesh, which comprises triangular faces. A user uses Markit to mark the triangular faces of elements in color. Each marked face is a hotspot that has textual information added by the maker. A user works on an imported model by following these sections one by one.

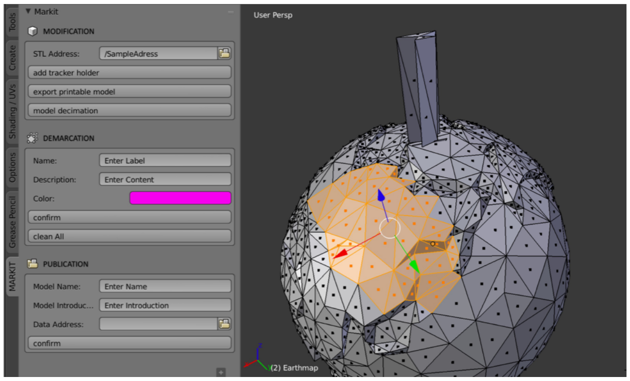
For an imported model, a user uses the functions in the Modification section to add a tracker scaffold, export printable model, and decimate the model. Markit loads a tracker scaffold when the user presses the first button in this section. Then, the user drags the holder to an appropriate position on the model, and presses the export button after entering the destination path of the export file. Since some complicated models with millions of 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 faces 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. In the prototype, the Publication section exports a Python pickle file.

Talkit

Talkit interprets augmented information from Markit, and 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.



STL Address: /SampleAdress
Name: Enter Label
Description: Enter Content
Model Name: Enter Name
Model Introduc... Enter Introduction

Figure 4. 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.

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 2b), slides the tracker (Figure 2c) into the printed scaffold, and enters a model ID (using the 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.

Algorithm Design

Locating a model. There are three coordinate systems involved in Talkit. In Markit, 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 Markit's data. Based on these coordinates, Talkit transforms all triangular faces into the world coordinate system. Talkit then maps the 3D world coordinate system into the projective coordinate system by recognizing the current position of the three markers on the tracker. After transforming all faces into the projective coordinate system, Talkit uses a Backface Culling algorithm [68] 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. We originally used skin-color detection to track the finger [54]. 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 (described in the previous section), Talkit uses a 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.

Talkit then finds model faces near the location of the fingertip. 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. As shown in Figure 5, 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." If the user asks for further information, the system will say "North America up, South America down."

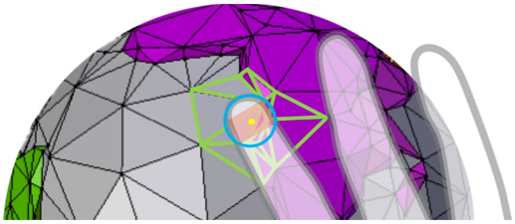


Figure 5. A user points her finger on the edge of the purple hotspot. The Talkit identifies the center point of the sticker (marked in yellow dot) as the position of the finger, and finds faces (outlined in green) that overlap with a circle (outlined in blue) around the center point.

Implementation

Talkit was implemented in Python. We used OpenCV [9] for real-time video processing. The Talkit application used the built-in calibration function from OpenCV [9] 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 [8] to recognize the markers and calculate the transformation matrix. For the speech input and audio guiding systems, we used pocketsphinx [24] to process speech input and used a text to speech engine to generate the audio output.

EVALUATING TALKIT

The goal of Talkit is to help blind users 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.

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 6, the four models included:

1. *Cell*: A cell model modified from thing: 689381 [39], 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 [67], the researcher marked five elements on the model. The model is symmetric, and there is "road wheel" on each side of the model.

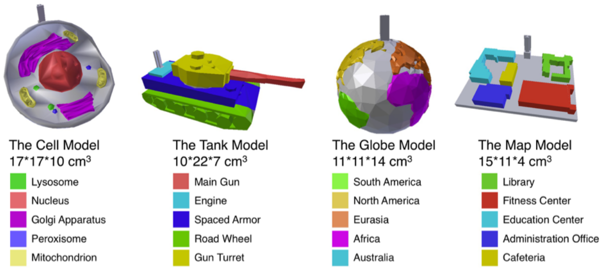


Figure 6. Four models used in the studies. The models and their elements are highlighted in color, and their names and dimensions are listed in the figure as well.

3. *Globe*: A globe model modified from thing: 17336 [6], the researcher annotated five continents.
4. *Map:* A map model downloaded from OSM Buildings [38]. 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, blind users 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' 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 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.” 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) [10] to collect subjective feedback from the participants. Table 1 shows the statements we used in the study. 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.

Results

The Time Needed to Access Annotations

In the first task, all participants found five elements on each model within three minutes (see Figure 7). 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 8 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 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,

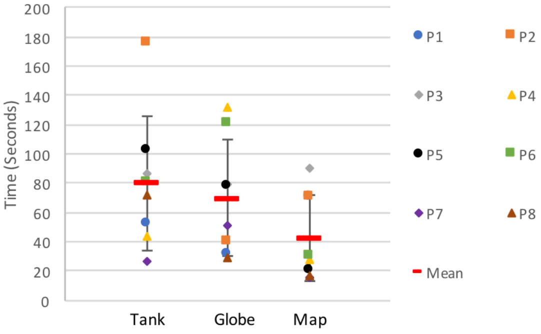


Figure 7. The time participants spent finding all five elements on each model. Mean times for all participants are shown in red lines.

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 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).

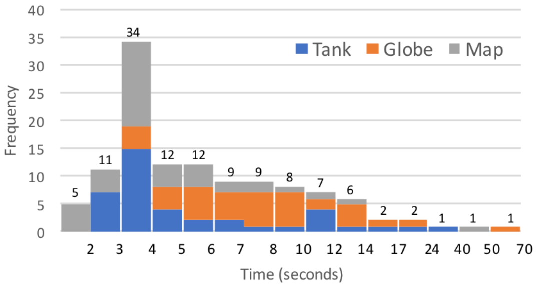


Figure 8. A histogram of the time participants took to locate an element, categorized and colored by models.

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

Table 1. The ten statements (S1 – S10) of the modified Standard Usability Scale (SUS) of the first 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.

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] 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 blind users 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.

EVALUATING MARKIT

The target users of Markit are "makers," 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.

Method

Participants

Eight sighted volunteers (4 male, 4 female) joined our study, denoted as P9-P16. Seven participants had at least college degrees, and one participant was pursuing her bachelor's degree. Five participants (P10-P14) had experience with modeling software but only one (P14) used modeling software in the last year, and four of them had experience with 3D printers except P11.

Apparatus and Materials

We ran Markit on a 13" 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

The study began with a training session. We introduced Blender and 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 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, 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.

We conducted a short interview with a modified SUS to collect subjective feedback at the end of this study.

Results

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 9 shows the mean times participants spent on different models. The Map model took less time than the others on average. P12 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 P15 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 subtasks, 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 textboxes 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.

Subjective Feedback

Table 2 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 (P9, P11, P12, P13) indicated that Markit could be improved in several ways (relatively negative scores for S2, S3, S5, S7, and S10). P12 and P13 attributed the difficulties they had with 3D navigation to the design of

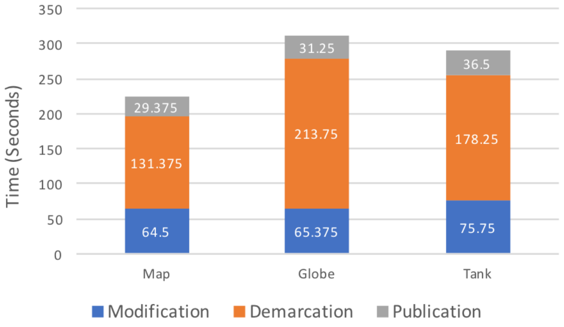


Figure 9. The mean time participants spent on each model, broken down by the different sections in Markit.

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. It's time consuming to create augmented models with this tool.							2.25
2	4	1		1			
S6. I think the information you can add with this tool is too limited.							2.625
1	2	4	1				
S8. The tool 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 tool.							2.75
1	2	4		1			

Table 2. The ten statements (S1 – S10) of the modified Standard Usability Scale (SUS) of the second 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.

Markit, and P12 emphasized that face selection was a "universal issue" that all 3D modeling software had. As P13 commented, "the fact that you have all these meshes you need to select, just ends up being time consuming." P9 shared the same feeling about the selection functions, but further explained, "the learning curve could be very sharp. Once you grab the sense of how to use different gestures, it could be very easy." P13 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, P16 suggested the Markit should provide indications if the input path or name is incorrect. P11 and P13 thought it would be nice to associate elements with other media like music.

DISCUSSION AND FUTURE WORK

Markit and Talkit achieved promising results in the user studies. 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. Blind participants 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.

The Size of Elements

One question both blind and sighted participants shared was how small could an annotated element be. In Markit, the size of an element depended on the size of its faces. Since a user can always subdivide a face using the Blender's built-in functions, the size of an element could be unlimited. In Talkit, 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 Physical Tracker

The current tracking system relies on a physical tracker, which requires additional edits in Markit and changes the tactile features in models. Prior work used self-contained color and shape information to track Duplo blocks [21] and paper puppets [5] without additional trackers. In the future, we will explore the feasibility of tracking 3D prints using self-contained information without the tracker.

Improving Talkit

The user input methods could be improved. Although the participants liked the robustness of the color sticker we used to track fingers, we think it's worthwhile to develop algorithms that can recognize multiple fingers without special stickers. In addition to gesture input, the set of speech commands could also be expanded. For example, we could implement a bot that can teach and talk to blind users.

Improving Markit

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 Bigger Picture

We believe that with learnable and usable tools, we can harness the enthusiasm of makers to create and share augmented models that will massively improve blind people's lives. Our *ultimate goal* is to build an *online community* that shares augmented models. We envision that makers will engage in the community and create augmented models using the tools we built, while blind people will be able to print models at home and learn from them about various concepts independently. Markit and Talkit represent the first building blocks towards achieving this vision.

However, several barriers still stand in the way. 3D modeling software still requires some technical expertise and practice. This was evident in the Markit evaluation, when tech-savvy participants with experience using 3D printing technology had some difficulty manipulating the 3D models in blender. At this point, it is unlikely that a teacher, for example, would be comfortable using Markit to annotate a model for her visually impaired student. As maker technology evolves, it will become more accessible to a wider range of people. Moreover, 3D printing is slow and 3D printers are not yet available in many homes or schools. Again, we believe this will change in the next few years, making the Markit and Talkit system more powerful.

As we have stated in this paper and prior work [53–55], our research is driven by the values of independence and empowerment; our interactive models are designed to help blind people explore and learn from models without relying on a knowledgeable sighted person to explain a model's elements. However, it is important to note that now maker technologies are generally not accessible. Designing 3D models and adding annotations with Markit are visually-demanding tasks. Thus, blind end-users are still dependent on sighted people, particularly those who have experience with maker technology. In the future, we plan to design an accessible method for adding annotations with gestures and speech input using a printed model. This added annotation approach will enable blind people to also contribute to the collaborative online community.

CONCLUSION

In this paper, we presented Markit and Talkit, components of a low-barrier toolkit that augmented 3D models with audio annotations. Markit enables makers to easily add audio annotations to online models, while Talkit allows blind people to access the annotations with a pointing gesture and speech commands. The user studies showed that Markit and Talkit were promising educational tools for blind people.

ACKNOWLEDGEMENT

This work was partially funded by AOL. We thank Yin Cui, Serge Belongie, and Lihi Zelnik for their help. We also thank all participants for their time and helpful feedback.

REFERENCES

1. Eric Akaoka, Tim Ginn, and Roel Vertegaal. 2010. DisplayObjects: Prototyping Functional Physical Interfaces on 3D Styrofoam, Paper or Cardboard Models. In *Proceedings of the fourth international conference on Tangible, embedded, and embodied interaction (TEI '10)*, 49–56. https://doi.org/10.1145/1709886.1709897
2. American Printing House. Product: DNA Twist. Retrieved April 2, 2017 from https://shop.aph.org/webapp/wcs/stores/servlet/Product_DNA Twist_1-08978-00P_10001_11051
3. American Printing House. Product: Globe: Tactile and Visual. Retrieved April 2, 2017 from https://shop.aph.org/webapp/wcs/stores/servlet/Product_Globe: Tactile and Visual_1-01551-00P_10001_11051
4. Catherine M. Baker, Lauren R. Milne, Jeffrey Scofield, Cynthia L. Bennett, and Richard E. Ladner. 2014. Tactile graphics with a voice. In *Proceedings of the 16th international ACM SIGACCESS conference on Computers & accessibility (ASSETS '14)*, 75–82. https://doi.org/10.1145/2661334.2661366
5. Connelly Barnes, David E. Jacobs, Jason Sanders, Dan B Goldman, Szymon Rusinkiewicz, Adam Finkelstein, and Maneesh Agrawala. 2008. Video Puppetry: A Performative Interface for Cutout Animation. *ACM Transactions on Graphics* 27, 5: 1–9. https://doi.org/10.1145/1409060.1409077
6. bld. Textured Earth - Thingiverse. Retrieved April 4, 2017 from http://www.thingiverse.com/thing:17336
7. Blender. Free and Open 3D Creation Software. Retrieved April 4, 2017 from https://www.blender.org/
8. Quentin Bonnard, Severin Lemaignan, Guillaume Zufferey, Andrea Mazzei, Sebastien Cuendet, Nan Li, Ayberk Ozgur, and Pierre Dillenbourg. 2013. Chilitags 2: Robust Fiducial Markers for Augmented Reality and Robotics.
9. Gary R. Bradski and Adrian. Kaehler. 2008. *Learning OpenCV : computer vision with the OpenCV library*. O'Reilly.
10. John Brooke. 1996. SUS-A quick and dirty usability scale. *Usability evaluation in industry* 189, 194: 4–7.
11. Craig Brown and Amy Hurst. 2012. VizTouch: automatically generated tactile visualizations of coordinate spaces. In *Proceedings of the Sixth International Conference on Tangible, Embedded and Embodied Interaction (TEI '12)*, 131–138. https://doi.org/10.1145/2148131.2148160
12. Emeline Brule, Gilles Bailly, Anke Brock, Frederic Valentin, Grégoire Denis, and Christophe Jouffrais. 2016. MapSense: Multi-Sensory Interactive Maps for Children Living with Visual Impairments. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*, 445–457. https://doi.org/10.1145/2858036.2858375
13. Erin Buehler, Stacy Branham, Abdullah Ali, Jeremy J Chang, Megan Kelly Hofmann, Amy Hurst, and Shaun K Kane. 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)*, 525–534. https://doi.org/10.1145/2702123.2702525
14. Erin Buehler, Niara Comrie, Megan Hofmann, Samantha McDonald, and Amy Hurst. 2016. Investigating the Implications of 3D Printing in Special Education. *ACM Transactions on Accessible Computing* 8, 3: 1–28. https://doi.org/10.1145/2870640
15. Erin Buehler, Amy Hurst, and Megan Hofmann. 2014. Coming to grips: 3D printing for accessibility. In *Proceedings of the 16th international ACM SIGACCESS conference on Computers & accessibility (ASSETS '14)*, 291–292. https://doi.org/10.1145/2661334.2661345
16. Erin Buehler, Shaun K. Kane, and Amy Hurst. 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)*, 107–114. https://doi.org/10.1145/2661334.2661365
17. Giovanni Fusco and Valerie S. Morash. 2015. The Tactile Graphics Helper : Providing Audio Clarification for Tactile Graphics Using Machine Vision Categories and Subject Descriptors. *Proceedings of the 17th International ACM SIGACCESS Conference on Computers & Accessibility (ASSETS '15)*: 491–500. https://doi.org/10.1145/2700648.2809868
18. Timo Götzelmann. 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)*, 81–90. https://doi.org/10.1145/2982142.2982163
19. Anhong Guo, Xiang "Anthony" Chen, Haoran Qi, Samuel White, Suman Ghosh, Chieko Asakawa, and Jeffrey P. Bigham. 2016. VizLens : A Robust and Interactive Screen Reader for Interfaces in the Real World. In *Proceedings of the 29th Annual Symposium*

on User Interface Software and Technology (UIST '16), 651–664. https://doi.org/10.1145/2984511.2984518

20. Anhong Guo, Jeeeun Kim, Xiang 'Anthony Chen, Tom Yeh, Scott E Hudson, Jennifer Mankoff, and Jeffrey P Bigham. 2017. Facade: Auto-generating Tactile Interfaces to Appliances. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17)*, 5826–5838. https://doi.org/10.1145/3025453.3025845

21. Ankit Gupta, Dieter Fox, Brian Curless, and Michael Cohen. 2012. DuploTrack: A Real-time System for Authoring and Guiding Duplo Block Assembly. In *Proceedings of the 25th annual ACM symposium on User interface software and technology (UIST '12)*, 389–402. https://doi.org/10.1145/2380116.2380167

22. Mark Hatch. 2013. *The Maker Movement Manifesto: Rules for Innovation in the New World of Crafters, Hackers, and Tinkerers.* McGraw Hill Professional. https://doi.org/10.1036/9780071821131

23. Megan Hofmann, Jeffrey Harris, Scott E. Hudson, and Jennifer Mankoff. 2016. Helping Hands: Requirements for a Prototyping Methodology for Upper-limb Prosthetics Users. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*, 1769–1780. https://doi.org/10.1145/2858036.2858340

24. David Huggins-daines, Mohit Kumar, Arthur Chan, Alan W Black, Mosur Ravishankar, and Alex I. Rudnicky. 2006. PocketSphinx: A free, real-time continuous speech recognition system for hand-held devices. In *Proceedings of ICASSP*. https://doi.org/10.1.1.67.2374

25. Amy Hurst and Jasmine Tobias. 2011. Empowering individuals with do-it-yourself assistive technology. *The proceedings of the 13th international ACM SIGACCESS conference on Computers and accessibility (ASSETS '11)*: 11–18. https://doi.org/10.1145/2049536.2049541

26. jaqtikkun. Cutaway Earth with Braille - Thingiverse. Retrieved April 1, 2017 from http://www.thingiverse.com/thing:510654

27. Shaun K. Kane and Jeffrey P. Bigham. 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)*, 247–252. https://doi.org/10.1145/2538862.2538975

28. Shaun K. Kane, Brian Frey, and Jacob O. Wobbrock. 2013. Access Lens: A Gesture-Based Screen Reader for Real-World Documents. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13)*, 347–350. https://doi.org/10.1145/2470654.2470704

29. Jeeeun Kim and Tom Yeh. 2015. Toward 3D-Printed Movable Tactile Pictures for Children al Impairments with Visual Impairments. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI '15)*, 2815–2824. https://doi.org/10.1145/2702123.2702144

30. Michael A. Kolitsky. Making Tactile 3D Prints Talk. Retrieved April 2, 2017 from http://www.instructables.com/id/Making-Tactile-3D-Prints-Talk/?ALLSTEPS

31. Michael A. Kolitsky. 2015. 3D printing makes Visible Human cadaver sections accessible for blind students. Retrieved April 2, 2017 from http://www.nextgenemedia.com/NLM3Dprints/NLM3Dprints.html

32. Gierad Laput, Eric Brockmeyer, Scott E. Hudson, and Chris Harrison. 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)*, 2161–2170. https://doi.org/10.1145/2702123.2702414

33. Make: Assistive technology News, Reviews and More. Retrieved April 4, 2017 from http://makezine.com/tag/assistive-technology/

34. Assistive Technology Makers. Helping Makers Help Others. Retrieved April 4, 2017 from http://atmakers.org/

35. Samantha McDonald, Joshua Dutterer, Ali Abdolrahmani, Shaun K. Kane, and Amy Hurst. 2014. Tactile aids for visually impaired graphical design education. In *Proceedings of the 16th international ACM SIGACCESS conference on Computers & accessibility (ASSETS '14)*, 275–276. https://doi.org/10.1145/2661334.2661392

36. Roderick Murray-Smith, John Williamson, Stephen Hughes, and Torben Quaade. 2008. Stane: Synthesized Surfaces for Tactile Input. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '08)*: 1299–1302. https://doi.org/10.1145/1357054.1357257

37. National Federation of the Blind Jernigan Institute. 2009. The Braille Literacy Crisis in America. *National Federation of the Blind*.

38. OSM Buildings. OSM Buildings. Retrieved April 4, 2017 from https://osmbuildings.org/

39. PaLEoS. Animal Cell - Thingiverse. Retrieved April 4, 2017 from http://www.thingiverse.com/thing:689381

40. Huaishu Peng, François Guimbretière, James Mccann,

and Scott E Hudson. 2016. A 3D Printer for Interactive Electromagnetic Devices. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology (UIST '16)*, 553–562. https://doi.org/10.1145/2984511.2984523

41. Huaishu Peng, Jennifer Mankoff, Scott E Hudson, and James McCann. 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)*, 1789–1798. https://doi.org/10.1145/2702123.2702327

42. Thomas Poon and Ronit Ovadia. 2008. Using Tactile Learning Aids for Students with Visual Impairments in a First-Semester Organic Chemistry Course. *Journal of Chemical Education* 85, 2: 240. https://doi.org/10.1021/ed085p240

43. Andreas Reichinger, Anton Fuhrmann, Stefan Maierhofer, and Werner Purgathofer. 2016. Gesture-Based Interactive Audio Guide on Tactile Reliefs. In *Proceedings of the 18th International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS '16)*, 91–100. https://doi.org/10.1145/2982142.2982176

44. roman_hegglin. Customizable Atom Deluxe (every element preconfigured) - Thingiverse. Retrieved April 1, 2017 from http://www.thingiverse.com/thing:114247

45. Audrey C. Rule. 2011. Tactile Earth and Space Science Materials for Students with Visual Impairments: Contours, Craters, Asteroids, and Features of Mars. *Journal of Geoscience Education* 59, 4: 205–218. https://doi.org/10.5408/1.3651404

46. Audrey C. Rule, Greg P. Stefanich, Robert M. Boody, and Belinda Peiffer. 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. https://doi.org/10.1080/09500693.2010.506619

47. Valkyrie Savage, Colin Chang, and Björn Hartmann. 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)*, 447–456. https://doi.org/10.1145/2501988.2501992

48. Valkyrie Savage, Sean Follmer, Jingyi Li, and Björn Hartmann. 2015. 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)*, 103–108. https://doi.org/10.1145/2807442.2807508

49. Valkyrie Savage, Andrew Head, Björn Hartmann, Dan B. Goldman, Gautham Mysore, and Wilmot Li. 2015. Lamello: Passive Acoustic Sensing for Tangible Input Components. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI '15)*, 1277–1280. https://doi.org/10.1145/2702123.2702207

50. Valkyrie Savage, Ryan Schmidt, Tovi Grossman, George Fitzmaurice, and Björn Hartmann. 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)*, 3–12. https://doi.org/10.1145/2642918.2647374

51. Martin Schmitz, Mohammadreza Khalilbeigi, Matthias Balwierz, Roman Lissermann, Max Mühlhäuser, and Jürgen Steimle. 2015. Capricate: A Fabrication Pipeline to Design and 3D Print Capacitive Touch Sensors for Interactive Objects. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology (UIST '15)*, 253–258. https://doi.org/10.1145/2807442.2807503

52. Huiying Shen, Owen Edwards, Joshua Miele, and James M. Coughlan. 2013. CamIO. In *Proceedings of the 15th International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS '13)*, 1–2. https://doi.org/10.1145/2513383.2513423

53. Lei Shi. 2015. Talkabel: A Labeling Method for 3D Printed Models. In *Proceedings of the 17th International ACM SIGACCESS Conference on Computers & Accessibility (ASSETS '15)*, 361–362. https://doi.org/10.1145/2700648.2811327

54. Lei Shi, Ross McLachlan, Yuhang Zhao, and Shiri Azenkot. 2016. Magic Touch: Interacting with 3D Printed Graphics. In *Proceedings of the 18th International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS '16)*, 329–330. https://doi.org/10.1145/2982142.2982153

55. Lei Shi, Idan Zelzer, Catherine Feng, and Shiri Azenkot. 2016. 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)*, 4896–4907. https://doi.org/10.1145/2858036.2858507

56. Lei Shi, Yuhang Zhao, and Shiri Azenkot. 2017. Designing Interactions for 3D Printed Models with Blind People. In *In Proceedings of the 19th International ACM SIGACCESS Conference on Computers & Accessibility (ASSETS '17)*. https://doi.org/10.1145/3132525.3132549

57. Roy Shilkrot, Jochen Huber, Wong Meng Ee, Pattie Maes, and Suranga Chandima Nanayakkara. 2015. FingerReader: A Wearable Device to Explore Printed Text on the Go. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing*

Systems (CHI '15), 2363–2372. https://doi.org/10.1145/2702123.2702421

58. Hyunyoung Song, François Guimbretière, Chang Hu, and Hod Lipson. 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)*, 1–10. https://doi.org/10.1145/1166253.1166258
59. Abigale Stangl, Jeeeun Kim, and Tom Yeh. 2014. 3D printed tactile picture books for children with visual impairments. In *Proceedings of the 2014 conference on Interaction design and children (IDC '14)*, 321–324. https://doi.org/10.1145/2593968.2610482
60. Lee Stearns, Ruofei Du, Uran Oh, Catherine Jou, Leah Findlater, David A. Ross, and Jon E. Froehlich. 2016. Evaluating Haptic and Auditory Directional Guidance to Assist Blind People in Reading Printed Text Using Finger-Mounted Cameras. *ACM Transactions on Accessible Computing* 9, 1: 1–38. https://doi.org/10.1145/2914793
61. Saiganesh Swaminathan, Thijs Roumen, Robert Kovacs, David Stangl, Stefanie Mueller, and Patrick Baudisch. 2016. Linespace: A Sensemaking Platform for the Blind. *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*: 2175–2185. https://doi.org/10.1145/2858036.2858245
62. Brandon Taylor, Anind Dey, Dan Siewiorek, and Asim Smailagic. 2016. Customizable 3D Printed Tactile Maps as Interactive Overlays. In *Proceedings of the 18th International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS '16)*, 71–79. https://doi.org/10.1145/2982142.2982167
63. Jolene K. Teske, Phyllis Gray, Mason A. Kuhn, Courtney K. Clausen, Latisha L. Smith, Sukainah A. Alsubia, Maryam Ghayoorad, Audrey C. Rule, and Jean Suchsland Schneider. 2014. Teacher-made Tactile Science Materials with Critical and Creative Thinking Activities for Learners including those with Visual Impairments. *Online Submission*.
64. TOM. TOM: Tikkun Olam Makers. Retrieved April 4, 2017 from http://tomglobal.org/
65. Marynel Vázquez, Eric Brockmeyer, Ruta Desai, Chris Harrison, and Scott E. Hudson. 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)*, 1295–1304. https://doi.org/10.1145/2702123.2702569
66. Karl Willis, Eric Brockmeyer, Scott Hudson, and Ivan Poupyrev. 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)*, 589–598. https://doi.org/10.1145/2380116.2380190
67. xehpuk. Simplified Tiger Tank - Thingiverse. Retrieved April 4, 2017 from http://www.thingiverse.com/thing:90265
68. Hansong Zhang and Kenneth E. Hoff. 1997. Fast backface culling using normal masks. In *Proceedings of the 1997 symposium on Interactive 3D graphics (SI3D '97)*, 103-. https://doi.org/10.1145/253284.253314
69. Yang Zhang, Gierad Laput, and Chris Harrison. 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)*, 1–14. https://doi.org/10.1145/3025453.3025842