Molder: An Accessible Design Tool for Tactile Maps

Lei Shi[1,2] **Yuhang Zhao**[1,2] **Ricardo Gonzalez Penuela**[1,3]
Elizabeth Kupferstein[1] **Shiri Azenkot**[1]
[1]Jacobs Technion-Cornell Institute, Cornell Tech, New York, NY, U.S.A.
[2]Information Science, Cornell University, Ithaca, NY, U.S.A.
[3]Systems and Computing Engineering, Universidad de Los Andes, Bogotá, Colombia
{ls776, yz769, reg255, ek544, shiri.azenkot}@cornell.edu

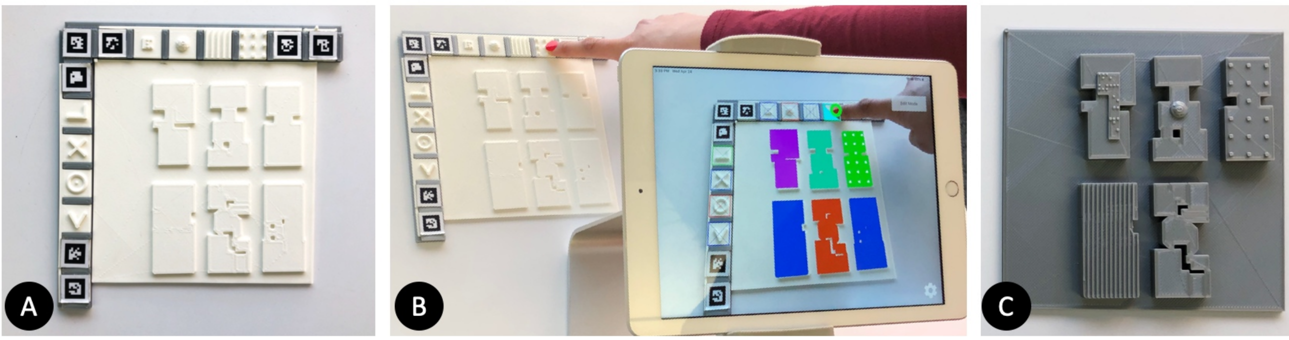

Figure 1. A teacher with visual impairments uses Molder to create an interactive tactile map for her students. (A) She prints a draft map model and places it in the Physical Ruler. (B) Then, the teacher uses her finger to select and modify buildings. The Molder application provides high-contrast visual feedback and auditory feedback to assist the teacher to modify the model. (C) When she finishes, she prints the final model, which is larger and has more instructive tactile features accompanying with interactive audio labels. The final model is more accessible and informative for teaching and learning purposes.

ABSTRACT
Tactile materials are powerful teaching aids for students with visual impairments (VIs). To design these materials, designers must use modeling applications, which have high learning curves and rely on visual feedback. Today, Orientation and Mobility (O&M) specialists and teachers are often responsible for designing these materials. However, most of them do not have professional modeling skills, and many are visually impaired themselves. To address this issue, we designed Molder, an accessible design tool for interactive tactile maps, an important type of tactile materials that can help students learn O&M skills. A designer uses Molder to design a map using tangible input techniques, and Molder provides auditory feedback and high-contrast visual feedback. We evaluated Molder with 12 participants (8 with VIs, 4 sighted). After a 30-minute training session, the participants were all able to use Molder to design maps with customized tactile and interactive information.

Author Keywords
Visual impairments; tactile maps; design tool

CHI '20, April 25–30, 2020, Honolulu, HI, USA

ACM 978-1-4503-6708-0/20/04…$15.00
https://doi.org/10.1145/3313831.3376431

CSS Concepts
• **Human-centered computing~Accessibility**

INTRODUCTION
Tactile materials are important learning tools for students with visual impairments (VIs). Models and tactile graphics (raised line drawings) are especially useful for teaching concepts that are too big, too small, or too dangerous to touch directly. For example, a Teacher of the Visually Impaired (TVI) typically uses models of molecules and other STEM concepts, while an Orientation and Mobility (O&M) specialist frequently uses models and tactile graphics to represent maps and other geographic information. These VI professionals often create tactile materials with arts and crafts. Such manual creation processes take time and effort and are limited by the maker's artistic ability.

Recently, VI professionals and educational institutions have started using 3D printers to create tactile materials for students with VIs. Instead of creating maps from scratch, digital designs can be shared among VI professionals and users. Printed materials are more robust than the ones made by hand with arts and crafts materials and can provide more detailed and accurate tactile features such as delicate tactile patterns and braille labels.

However, most VI professionals are not trained as professional designers, and 3D modeling tools are complex and have a steep learning curve. 3D modeling software, like Rhino [42] and Blender [5], require a user to manipulate a 3D shape through a 2D interface. This requires good visuospatial reasoning and can take hours of practice to learn. In addition, these 3D modeling tools are designed to support designers of any 3D models, and do not have functions that support tactile materials for people with VIs specifically. As a result, certain basic operations in the context of VIs, like adding a braille label to a model, are complex and involve many operations. Each braille dot must be created and placed individually to form letters and words on a model.

Moreover, 3D modeling interfaces are not accessible to people with VIs. They use a graphical user interface (GUI) and a mouse, revolving around the 2D visualization of the 3D shape. Since many VI professionals themselves have visual impairments, this is a major barrier. Recently, script-based modelers like OpenSCAD [36] make 3D modeling accessible to people with VIs but require a foundation in programming.

To address these two major difficulties of 3D modeling software, we envision a tactile modeling tool that includes a set of functions that are commonly used by VI professionals and uses accessible interaction techniques. For example, instead of creating individual dots to form a braille label, a user can perform one operation and specify the contents of the label. Instead of manipulating a 2D rendering of the 3D model, the user can interact with a 3D, or tactile, rendering of the model directly. Working towards this vision, we began by focusing on interactive tactile maps, an indispensable tool for daily mobility and orientation.

In this paper, we present Molder, an accessible design tool for interactive tactile maps. To simplify the design process, Molder includes functions commonly used for designing tactile maps, including adding braille and audio labels, adding textures and icons, removing geographic elements to reduce clutter, and scaling. To enable accessible design, Molder users perform these functions by directly manipulating a (tangible) draft model and accessories. In addition, users also interact with Molder on an accessible screen-based interface. The Molder application runs on an iOS device, using the camera to capture the draft model and the user's gestures. While Molder can be used nonvisually, the GUI has high-contrast colors making it accessible to people with low vision and full sight as well (see Figure 1B).

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 sessions, and designed a tactile map with braille labels, tactile patterns, and interactive audio 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 successfully enabled non-expert designers with VIs to design interactive tactile maps. Participants also suggested ways to make Molder easier to use and learn, like adding an interactive tutorial.

In summary, we contribute Molder, the first accessible design tool for tactile maps. While its functions are specific to tactile map design, Molder represents a new paradigm for 3D design that centers around genre-specific functions and a tangible interface. Moving forward, this approach can be adapted for other types of tactile materials.

RELATED WORK

In addition to tactile maps, people with visual impairments benefit from tactile materials of different kinds, such as tactile graphics (raised line drawings) and 3D models. Nowadays, people have started using 3D printers to print such tactile materials. While Molder supports the design of printed tactile maps in particular, it represents a paradigm for designing other printed tactile materials as well.

Printed and Interactive Tactile Materials

For people with VIs, 3D printed tactile materials are powerful tools in education [14,15]. Researchers have designed various printed tactile models and graphics for people with VIs [22–25,33]. For example, Stangl et al. [56] and Kim et al. [31] created printed tactile books to help children with VIs learn literacy skills. Kane and Bigham [30] used 3D printing to allow blind students to learn basic concepts of visualizations and programming. Guo et al. [26] used printed labels to make the interfaces of appliances more accessible.

Seeing the potential of 3D printed tactile materials, researchers augmented 3D models with interactive audio labels to convey more information [8,10,11,16,17]. Traditionally, braille labels are added to tactile materials, but the length of the labels is limited by the size and topology of a 3D model [9,27,29]. Thus, researchers created interactive models that could sense a user's behavior and provide descriptive audio labels to help the user understand the models. To sense a user's behavior on a printed model, researchers used different sensing techniques like acoustic sensing [48,51], embedding capacitive sensors into models [27,32], combining conductive printing materials with touchscreens [19,20,45,59], and computer vision [41,46,49,50,52]. For example, Shi et al.'s Talkit++ [49] was a mobile application for interactive models. It recognized the position of a printed model and a user's fingers using computer vision techniques. When a user pointed to a labeled component on the model, the application spoke the associated label. While some of these techniques could be applied to a variety of tactile materials, most prior work focused on augmenting tactile maps [1,8–11,16,17,27,29].

Creating Printed Tactile Materials

While TVIs and O&M specialist are responsible for creating tactile materials, studies show that they do not have training in 3D design and, as a result, they face challenges in designing tactile materials for people with VIs [18,43,47,55]. For example, Sheppard and Aldrich [47] found that producing tactile materials for blind students was labor intensive for TVIs. Moreover, Stangl et al. [55] found that VI professionals had difficulty designing models using SketchUp [54], a common 3D modeling application, and sought help from their peers and researchers.

To address the challenges experienced by non-expert designers including VI professionals, researchers designed technologies to lower the barrier of creating tactile models. Most researchers developed methods to automatically generate printable tactile materials from 2D images, especially 2D maps [21,37,60,65]. For example, Taylor et al. [60] developed a web tool that allowed visually impaired users to specify a location on an online map. The tool then generated a tactile map model of the specified location. VizTouch software [13] automatically generated tactile mathematical graphics from equations or csv files. However, these tools involved computer generated tactile models, without enabling education experts (e.g., TVIs) to create and modify their own models. Education experts must often customize tactile materials to meet the different needs of their students and clients, so these tools would be of limited use to them.

Instead of using computer generated content, Swaminathan et al. [57] took another approach. They built a large tactile display system (140x100cm) that enabled 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 was a sculpted element attached to a solid background of the same material. Using the system, a user created drawings using gestures (e.g., pointing) and speech commands (e.g., "line"). However, these designed tactile graphics were fixed to the large system and could not be distributed to users as tactile maps.

In addition to design tools for traditional tactile materials, some researchers proposed design tools for interactive tactile models. For example, Shi et al. [52] augmented 3D models with interactive audio labels. They created Markit, a design tool that allowed a maker to add interactive audio labels to a 3D model. These labels can be accessed through a mobile application. However, Markit was designed for sighted people with 3D modeling skills and relied heavily on visual feedback. Thus, TVIs and users with VIs would have difficulty using it.

Our work is the first to present a modeling design tool that are accessible for people with VIs and simple enough for non-expert model designers to use. Molder can automatically generate draft tactile maps and allows a non-expert designer with VIs to easily modify these generated maps. The designer can also add interactive audio labels to different buildings on the maps. With the help of tangible interaction, auditory feedback, and high-contrast visual feedback, Molder can be operated by designers with different visual abilities.

Tangible Modeling Tools

In prior work, researchers designed tangible 3D modeling tools for sighted designers by leveraging physical objects. MixFab [66] was a mixed reality environment for personal fabrication, where a user could interact with a virtual model with gestures and introduce existing physical objects into the model design. Window-Shaping [28] was a mobile application that allowed users to create digital 3D shapes directly on and around physical objects. Peng et al. designed RoMA [39], which allowed a designer to design and modify a 3D model digitally via an augmented reality headset and a controller, while a robot arm printed the model simultaneously. Ether-Toolbars [40] enabled designers to use physical papers as off-screen toolbars. However, these 3D modeling tools largely relied on visual feedback, overlooking the needs of people with VIs. To help designers with VIs to design physical objects, ShapeCAD [53] used a 2.5D tactile shape display, which was expensive and impractical for most users. To our knowledge, Molder is the first accessible and low-cost design tool for tactile maps. While it focuses on features specific to tactile maps, it presents a new accessible paradigm and can be adapted for designing different types of tactile materials.

DESIGN GOALS

To form our design goals, we first sought to answer several questions: What are the design practices of expert and non-expert designers of tactile materials? What challenges do they face? What are important considerations for making tactile models, especially maps, effective? To answer these questions, we conducted a formative study and distilled key insights from prior work.

Formative Study: Practices of Tactile Material Designers

To understand the practices, challenges, and considerations of tactile material designers, we joined an online special interest group about 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 whom were not professional model designers.

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

We found that the designers started their design process with a draft model. The draft model was typically downloaded from online resources (e.g., Thingiverse [61] or TouchMapper [63]) or converted from a 2D sketch (e.g., TactileView [58]). Some draft models were in stereolithography (STL) format and were therefore difficult to edit in a modeling software application.

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 [62], SketchUp [54], Blender [5], and Rhino [42]. 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 were printable and big enough to touch. For overly complicated models, the designers deleted and rearranged model components to simplify the layout.

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. For example, one model designer, who was a low vision teacher, told us that she worked with a makerspace to produce models for her students. This resulted in a long turnaround time, however, which led to the models being no longer relevant or needed.

Reviewing Prior Work: Effective Tactile Maps

In prior work, researchers used braille labels and tactile patterns to annotate tactile maps. In addition to braille labels [21,23,30], there were two types of common tactile patterns. One type was a single primitive like a circular cone or a cube [23,25]. The other one was filling an element with repeated primitives. For example, Taylor et al. [60] used repeated domes to represent waterways, and filled another area with pyramids to represent a park.

Researchers also added interactive audio labels to tactile maps [1,8–11,16,17,27,29,50,51]. As discussed in the Related Work, 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 interactive maps enable people with VIs to quickly retrieve rich map information and are accessible to people who do not know braille.

The Derived Design Goals

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

1. Molder should allow designers to iterate on a draft model.
2. Molder should support common modeling functions.
3. Molder should enable designers to add audio labels.
4. Molder should be easy to learn and easy to use by nonexperts.
5. Molder should be accessible to users with VIs.

THE DESIGN OF MOLDER

Molder addresses two critical issues in model design. First, traditional modeling tools are inaccessible to people with VIs. Instead of using a mouse and a monitor as input and output interfaces, Molder uses tangible input techniques and provides accessible visual and auditory feedback. Second, traditional modeling tools are built for professional designers and have steep learning curves for non-expert designers. When designing Molder, we facilitated a simplified design process based on our observations from the formative study and prior work.

Design Process

Creating an interactive map model involves different skills, and Molder simplifies this process. The simplified design process consists of three major parts: (1) creating a draft model through the Molder website, (2) modifying the model using physical tools and the Molder application, and (3) allowing end users (e.g., students with VIs) to interact with the model using the Molder application.

A designer starts the design process by generating a draft model. This is done by entering geographic information for a desired region on the Molder website. Using the entered information, Molder automatically generates the draft model, which is ready for a designer to print with a 3D printer.

The designer then launches the Molder application and edits the model. She selects elements and performs functions with the help of tangible input techniques and accessible feedback. Her edits can include modifying the tactile features of the model, like adding braille labels and deleting buildings, and also adding interactive audio labels to the different geographic components on the model.

Once the designer finishes her edits, she can print the edited model for her students. The Molder application has an interaction mode that allows students to explore the model with audio labels. In the interaction mode, the student can briefly touch a building to retrieve its audio label. The Molder application pulls label data and renders audio output.

Tangible Input Techniques

Manipulating 3D models is a challenging task. In prior work, Shi et al. found that non-expert designers lacked spatial navigation skills and had difficulty controlling a 3D model on 2D GUIs [52]. As such, we designed tangible input techniques that allow designers to edit a model without relying on vision or spatial reasoning.

When using Molder, a designer performs gestures on physical tools to modify a draft model. A designer puts the printed draft model on The Physical Ruler (Figure 2), which offers a reference frame and allows the Molder application to track the position of the model. To modify the model, the designer uses his finger to select elements (i.e., buildings) and perform different functions. The functions are designed as two Tangible Widgets (Figure 3). When resizing a model, a designer uses The Indicator (Figure 2) to indicate the size.

To support these tangible input techniques, the Molder application tracks the user's gestures and the physical tools using a camera. A designer must put a red sticker on one of his fingers, which allows Molder to track the position of the finger. Then, he can briefly touch an element or a function to get its label, while holding his finger in place longer will select the element or the function. As seen in the figures, Molder uses visual markers to locate the physical tools.

While gesturing on the physical tools is used to perform the primary set of commands, the Molder application also allows designers to use touchscreen gestures and speech input to enter the content of braille labels and interactive audio labels.

Accessible Feedback
While traditional modeling tools render the feedback throughout the modeling process visually, Molder uses accessible multimodal feedback. First, the Molder application has auditory feedback that reports each selected element and function, as well as the current modeling status. Second, the application provides high-contrast visual feedback to highlight important information.

For auditory feedback, the Molder application speaks all key steps in the modeling process (e.g., selecting an element, performing a function). When the designer briefly touches a building, the application reports the building's unique identifier (ID). In addition, she can hold her finger over the building to select the building and get detailed information. For example, when the designer selects a building, Molder might report: "2 selected, has a braille label of 'library,' no label, no pattern." This indicates that the selected building's ID is 2 and has a braille label but does not have an audio label or tactile pattern. The application also notifies designers when the model is out of the camera's view.

For low-vision designers, Molder also provides visual feedback. The application highlights the currently selected building on the iOS device, and displays whatever modifications the user makes to the model. For example, if the user adds a tactile pattern to a building, the application overlays the tactile pattern on the building displayed on the screen, showing how it will appear in the final model.

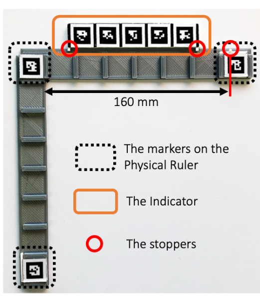

Figure 2. The Physical Ruler and The Indicator. There are three markers and one stopper on the Physical Ruler. The Indicator has two stoppers and a rigid tactile pattern on its right side.

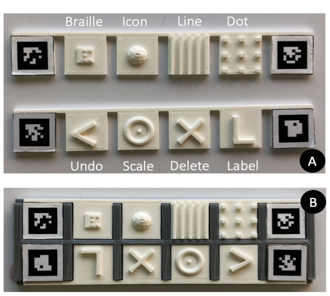

Figure 3. The design of the Tangible Widgets. (A) The top Tangible Widget has four functions: Braille, Icon, Line, and Dot. The Tangible Widget on the bottom has another four functions: Undo, Scale, Delete, and Label. (B) Designers can combine the two Tangible Widgets into a palette.

Functions
Traditional modeling applications are powerful tools used to design any 3D model. As a result, many operations must be performed to complete a single design task, like adding a braille label. To alleviate this complexity, the Molder paradigm involves incorporating first-order functions that are common to the tactile map genre.

Deletion
In the formative study, designers expressed their concerns about overcomplicated tactile information. If a model is cluttered with buildings that do not serve a particular purpose, a designer may want to delete these buildings. To perform deletion in Molder, the designer selects an unnecessary building, and chooses the Delete function. Because a deleted building still exists on the printed draft model, Molder reports, "no element, already deleted" when the designer tries to select the deleted building again.

Tactile Patterns
Tactile patterns help end users locate the buildings on a map. There are three types of tactile patterns supported by Molder: Icon, Line, and Dot. We chose these patterns because they were often used by the participants in the formative study and in prior work. A designer selects a building and then chooses the pattern he wishes to apply to it. Figure 4 shows the three patterns on the same building.

Braille Labels
To add a braille label, a designer selects a building and activates the Braille function, as shown in Figure 5. Then, she enters the content of the label in a textbox on the iOS device. The designer can use either the keyboard or dictation to enter content.

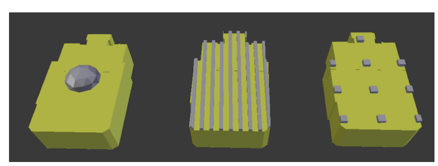

Figure 4. Three tactile patterns on the same building. From left to right, the added tactile patterns are Icon, Line, and Dot. The building is marked in yellow and the added patterns are marked in grey.

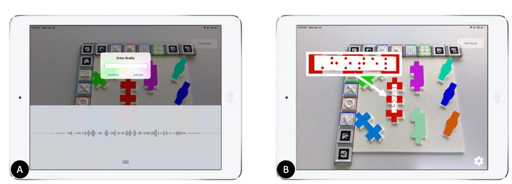

Figure 5. To add a braille label, a designer selects a building and the Braille function, and (A) enter information on a popup textbox. (B) Molder processes the command and updates the model. The braille label entered here is "Hotel."

Audio Labels
Since braille labels are limited by their size, audio labels can offer an important alternative. They contain more information and are accessible to those who don't read braille. To add an audio label, the designer selects a building and activates the Label function. Then, she enters the contents of the label in a textbox on the iOS device.

Scaling
Designers need to resize models to make sure their size is appropriate for tactile exploration. Unlike other functions that only require selection and text entry, the Scaling function requires a designer to indicate a desired size. Instead of asking the designer to enter a digital number, we follow our tangible design paradigm and designed the Indicator, a rectangular widget that is placed against the Physical Ruler. The right end of the Indicator stands for the final width of the model. To make the model smaller, she needs to move the Indicator to the left (Figure 6A). To make the model bigger, she needs to move the Indicator to the right (Figure 6B). Then, she selects the Scale function to resize the model.

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., reapply a braille label to a building). The designer can also use the Undo function to cancel the most recent command. For example, if the designer accidentally deletes building 3, she can hold the Undo function to recover the building.

IMPLEMENTATION
Molder consists of four components: (1) physical tools that are a part of the tangible input techniques, (2) an iOS application that designers interact with when designing a model, (3) a website that creates draft models, and (4) a server that handles model data.

Physical Tools
We designed the physical tools using Rhino and added visual markers on them. 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. 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, see Figure 2). All these physical tools are 3D printable and reusable.

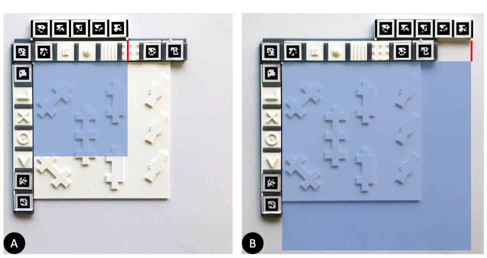

Figure 6. A designer can use the Indicator to change the size of a model. (A) To make the model smaller, she moves the Indicator to the left. (B) To make the model bigger, she moves the Indicator to the right. The red lines in the images stand for the right end of the Indicator. The blue rectangles represent the final sizes of the model.

The Molder Application
The Molder application senses input (i.e., tracking physical tools and gestures, detecting speech commands) and renders feedback for designers. In addition, end users (e.g., students with VIs) can also use the application to retrieve the audio labels. The application is implemented on the iOS platform, and is compatible with VoiceOver, the screen reader on iOS.

The application uses the Chilitags library [6] to locate the markers on the physical tools. Once it locates the markers, the application uses spatial information and transformation matrices to determine the positions of the tools and the model.

The application tracks a 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 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. 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. We set these thresholds empirically and intended to help designers avoid accidental operations.

In addition to tracking the physical tools and gestures, the application uses SpeechRecognizer [3] to detect speech commands.

Once it identifies an operation (e.g., deleting a building), the application sends a command to the Molder server and renders feedback with updated model data. The Molder application does not process model data locally. Instead, it sends commands to the Molder server, which modifies the model accordingly. Then, the Molder application renders auditory and visual feedback to the designer using a text-to-speech engine [4] and OpenCV[7].

The Molder Website
The website helps designers create draft models (Figure 7). We developed the website using HTTP, CSS, and JavaScript. The website takes geographic information (i.e., Nominatim) as input and outputs an STL model generated by the Molder

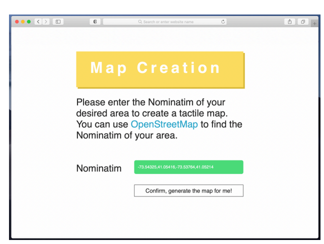

Figure 7. The user interface of the Molder website, which is used for creating draft models.

server to be used as a draft model. The Nominatim can be obtained by entering an address or a zip code in OpenStreetMap [38].

The Molder Server

The Molder server is in charge of processing model data, and is implemented as an add-on to Blender [5], a modeling software that includes basic modeling functions (e.g., Transformation, Boolean Modifiers). The user interface of the Molder server is shown in Figure 8. Although the server can be used as a standalone design tool, its inaccessible user interface is not compatible with screen readers. Thus, it acts as a server that takes commands from two accessible interfaces: the Molder application and the Molder website. The two accessible user interfaces send commands to the server and receive updated data from the server.

In this section, we will describe how the server handles model creation and model editing.

Model Creation

With the user input from the Molder website, the server creates a draft model. The server pulls 2D building outlines from OpenStreetMap [38]. Molder only imports 2D building outlines using Blender OSM [64] (Figure 9A).

Then, the server converts the outlines to a draft map model. OpenStreetMap may register a single building as several units (e.g., a building with a dozen shops). We combine these overlapping units to avoid overcomplicated information. The server also assigns high-contrast colors and IDs to each building to distinguish them from one another (Figure 9B).

Third, the 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 $160 \times 160 \times 2$ mm^3. We also add a $2 \times 2 \times 2$ mm^3 cube on the left top corner of the base to indicate the orientation of the model (Figure 9C).

Last, the 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 (Figure 9D) serve as an unedited model. All edits will be performed on the regular draft model. We set these heights empirically. In the future, designers should be able to customize the height of the regular model. Figure 10 shows a shallow draft model, an unedited regular draft model, and a final model.

Model Editing

The server takes commands from the Molder application, and edits models accordingly.

Deletion: Each building is an independent mesh, and we use Blender's functions to delete the selected building.

Tactile Patterns: For the Icon pattern, the server creates an icosphere at the center of a selected building. For the Line pattern and the Dot pattern, we fill the selected building with rectangle or dot meshes.

Braille Labels: The server converts English characters into a printable braille label using similar algorithms implemented in braille-printer [35]. In the current design, we only implemented grade 1 braille.

Audio Labels: The server stores audio labels as self-defined objects, which are readable to the Molder application.

Scaling: The server resizes the model with a user-define value. When resizing, the server keeps the original size of braille labels since braille has a standard size.

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

EVALUATION

To evaluate Molder, we conducted a study with blind, low vision, and sighted participants. Our goals were to assess the

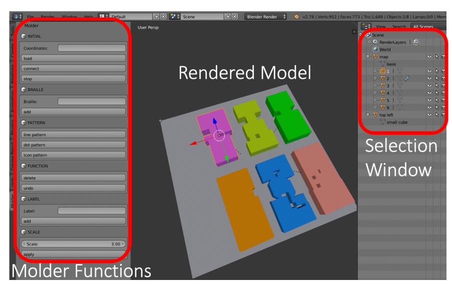

Figure 8. The user interface of the Molder server. A user can select a building in the selection window and perform Molder functions. The model will be updated in real time.

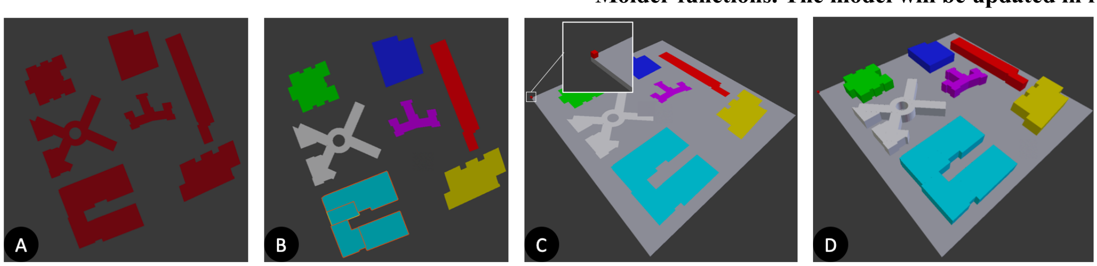

Figure 9. Molder automatically creates a draft model with the geographic information provided by a designer. (A) First, Molder imports 2D building outlines. (B) Then, it combines overlapping units to a single building (e.g., the aqua building has four units) and assigns high-contrast colors and IDs to all buildings. (C) Molder also adds a base (marked in grey) and a small cube (marked in red) on the top left corner to indicate the orientation of the model. (D) Last, Molder extrudes the 2D building outlines to 3D meshes.

usability of the Molder functions and the overall experience of creating maps.

Method

Participants

We recruited 12 participants (mean age = 57.5, SD = 16). Two participants identified as male, nine identified as female, and one identified as non-binary. We recruited participants through fliers at low vision community centers and email blasts through schools for TVIs.

Eight visually impaired participants, denoted as P1 - P8, participated in the study. Four participants identified as blind (P3 - P5 and P8) and four participants identified as low vision (P1, P2, P6 and P7). All participants had experience using and designing tactile materials. Most of them acquired their design experience by helping other designers create tactile materials like tactile maps of subway systems. Only P4 had used a 3D modeling tool (i.e., OpenSCAD) before the study. All of them were familiar with braille, and seven of them were familiar with the VoiceOver screen reader. None of them had enough vision to use the Molder server directly.

In addition, four sighted participants took part in the 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.

Apparatus

We set up the Molder server on a Macbook and ran the Molder application on an iPad. The iPad was placed in front of a participant on a stand (Figure 1B). We used the back camera of the iPad to track objects and added a wide-angle lens [2] to enhance the camera's field of view.

We prepared three draft models, denoted as Model A - Model C (Figure 11). Model A had 6 buildings, Model B had 8 buildings, and Model C had 12 buildings. To demonstrate Molder to participants, we modified Model A and printed the modified model, as shown in Figure 10.

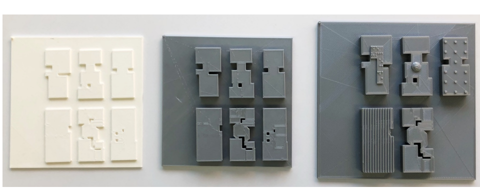

Figure 10. From left to right: a shallow draft model, an unedited regular draft model, and a modified model.

Figure 11. Three draft models we used in the study. From left to right: Model A, Model B, and Model C.

Procedure

The study consisted of a single session with each participant that lasted around 1.5 hours. First, we introduced Molder to participants. The participants learned how to use the tangible user interface (TUI), which included the physical tools, functions, and the Molder application. For sighted participants, we also introduced the GUI (i.e., the Molder server) as a standalone design tool. We used Model A and its modified model in the introduction. This process took around 30 minutes. Then the participants did 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 12. We announced the tasks one by one. The participants performed the announced task and spoke "done" when they finished the task. VI participants only used the TUI, while sighted participants used both TUI and GUI to perform the same tasks on Model B twice. We counterbalanced the order of the 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 way. VI participants used the TUI, while sighted participants could choose one of the two interfaces.

Last, we wrapped up the study with a short interview, where we used a questionnaire based on the Standard Usability Scale (SUS) [12] to evaluate the user experience of the TUI. We modified the SUS to make it relevant to Molder, as shown in Table 1. To reduce acquiescence and response biases, we mixed positive and negative statements.

Data Collection and Analysis

We video recorded all sessions, and two researchers developed themes from the recording transcripts using axial coding [44]. In addition, we calculated the time participants spent on each task in the structured design test and analyzed the modeling activities they had in the freelance design test.

Results

The Time Participants Spent in the Structured Design Test

All participants managed to finish T1 - T9 using the 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 13.

Across all participants and user interfaces, participants spent 22.47 (SD = 18.83) seconds on each task. On the TUI, VI

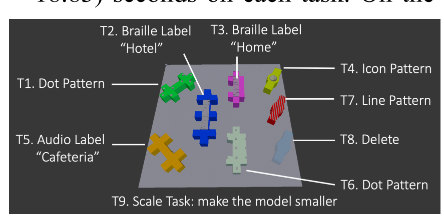

Figure 12. The nine tasks in the Structured Design Test.

participants (i.e., P1 - P8) spent 32.18 (SD = 21.83) seconds on each task, while sighted participants spent 16.83 (SD = 5.27) seconds on each task. On the 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 tasks (i.e., T1 - T9) and vision conditions (i.e., visually impaired and sighted) as independent variables. On average, we found that the participants with VIs spent 15.35 seconds more than sighted participants ($p < 0.01$). We observed that VI participants needed to find buildings and functions through tactile exploration, while sighted participants could quickly locate the desired building. The time spent by VI participants was also more varied (higher standard deviation). We think this may relate to their different visual abilities. Among all VI participants, P6 spent the longest time on average (mean = 60.7 seconds, SD = 26.13). She had long 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 the GUI compared to their performance on the TUI. In another ANOVA analysis, where we took the tasks and the interfaces (i.e., the GUI and the TUI) as variables. We found that sighted participants spent significantly more time on TUI compared to GUI ($p < 0.01$). When using the TUI, participants needed to hold their fingers to select an object. Recall that the time thresholds for selecting a building and a function were 1.5 seconds and 3.5 seconds respectively. When using the GUI, they selected an object without a time threshold.

The Modeling Activities in the Freelance Design Test

In Figure 14, 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 VI participant completed 9.75 (SD = 3.69) activities, and each sighted participant completed 13.5 (SD = 2.38) activities.

All participants used braille labels and nine participants used interactive 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 interactive audio labels (P6, P10, and P11) were not sure about the use cases of these. Other participants showed different preference between braille labels and interactive audio labels. P1 thought that "braille labels [should be] for more important things" and interactive audio labels should be used for orientation. Four participants (P2, P3, P8, and P9) said their choices were arbitrary because they wanted to try both types of labels. Two participants (P4 and P5) would like to add the two types of labels with the same content to keep consistency. P11 and P12 preferred interactive audio labels because "not everyone would know how to use braille."

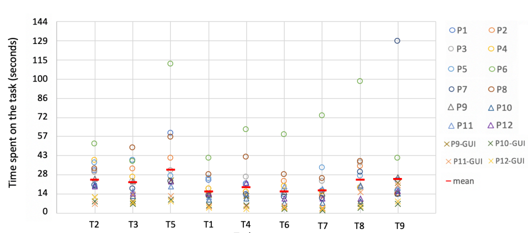

Figure 13. The time participants spent on each task. T2, T3, and T5 required text entry. T1, T4, T6, and T7 are related to different tactile patterns. T8 is a deletion task and T9 is a scaling task.

Subjective Feedback

Overall, participants highly rated Molder. Table 1 shows the responses to the modified SUS. The participants thought Molder could help them design tactile maps (high scores for S1 and S5). However, P10 thought the TUI was less useful compared to the GUI (relatively low 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 (low scores for S2, S4 and high scores for 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.

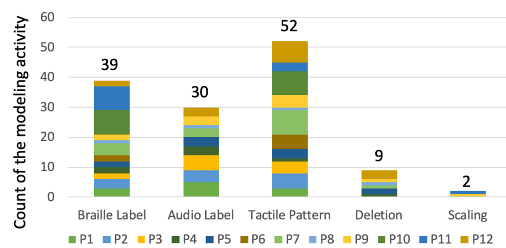

Figure 14. The count of the modeling activities in the freelance test.

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

Table 1. The ten statements (S1 – S10) of the questionnaire in the study, with their histograms and means scores. Each statement was scored from 1 (which indicates "strongly disagree") to 7 (which indicates "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 managed to learn Molder and performed different tasks in a short amount of time, but some of them thought the learning curve for using Molder was steep (relatively low scores for S7 and high scores for S10). Although the participants had experience in design and tactile materials, modeling was still a new task for many participants. P5 said, "I wouldn't say that it's necessarily hard, just different because it's new." This was echoed by P2, P3, P9, and P12. P6 and P12 were also concerned about VI people with neuropathy, who might not be able to use Molder because of the loss of tactile sensitivity. Instead of relying on personal instructions, P8 suggested that a tutoring mode in Molder could improve its learnability.

Some participants thought Molder could have more functions (relatively high scores for S6). For example, P4, P8, and P9 wanted to have more tactile patterns. P2 and P8 would like to add different layers of audio labels so users can get more detailed information if they wanted. P2 and P7 wanted to add new objects (e.g., stairs, benches) on draft models. In addition to braille labels, P10 wanted printed labels for people who do not know braille.

DISCUSSION AND FUTURE WORK

Molder enables non-expert designers to create and modify tactile maps. In the study, participants learned how to use Molder in about 30 minutes, which showed that Molder was easy-to-learn for non-expert designers. In addition, all participants managed to design maps, which showed that Molder's tangible input techniques and multimodal feedback were accessible to designers with different vision abilities. The study also exposed some limitations of Molder, which led to our thoughts on future work.

Tangible Modeling

In the study, both VI and sighted participants could use the Molder TUI to finish modeling tasks, although VI participants spent more time as they needed to find objects through tactile exploration. In the current implementation, Molder supports selection using only one finger, which limits the tactile exploration process. In the future, we will consider two-hand and multi-finger interactions.

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

While the TUI provided additional tactile feedback to help designers compose their ideas, we acknowledge that the cost of using the TUI was not negligible. In the future, we will also explore solutions that do not rely on printed accessories. For example, we can utilize a touchscreen from a smart device and vibration motors to enable a designer with a VI to select and modify elements of a tactile map.

Fabrication Methods

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

In this work, we designed the models and accessories in relatively small sizes considering the limited build volume of mainstream 3D printers, but it is possible to print large-scale models using special techniques. One alternative solution for fabricating a large tactile map is to break down the large map into several smaller printable, editable pieces.

Learnability

In the study, the researchers taught the participants how to use Molder in 30 minutes. Some participants found it challenging to master Molder in this limited amount of time. In the future, we should deploy a tutoring application embedded in Molder, which would enable designers to learn how to use Molder independently in the long term.

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 our 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., [34]) or use crowd-powered systems. For example, a designer can download a chemical molecule model online. Then, she can upload this model to a crowd-powered system to segment it. After the crowd workers segment the model into logical components (e.g., individual atoms), the designer can select and modify the atoms.

The designer can use similar functions and interaction techniques from Molder to modify the segmented model. While the functions and physical tools presented in this paper are designed for interactive maps, most of them can be applied to other types of tactile materials. For example, prior work added interactive audio labels to other types of models [52]. In addition, scaling and the Indicator are generally useful in 3D modeling. We also expect that future researchers and designers will contribute new functions and physical tools that can enrich the functionalities of Molder.

CONCLUSION

We presented Molder, an accessible design tool for interactive tactile maps. Molder provides (1) a simplified design process with a set of functions, (2) tangible input techniques, and (3) accessible feedback to help designers with different visual abilities create maps. We evaluated Molder in a study with 12 visually impaired and sighted participants. The study showed that Molder was accessible and useful, and yielded design implications to improve the application. As such, Molder presents an important first step and a new interaction paradigm in accessible modeling.

ACKNOWLEDGEMENTS

This work was supported in part by a gift from Verizon Media and by the National Science Foundation, under Grant No. 1746123.

REFERENCES

[1] Jérémy Albouys-Perrois, Jérémy Laviole, Carine Briant, and Anke Brock. 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), 1–14. https://doi.org/10.1145/3173574.3174203

[2] Amazon.com. Xenvo Pro Lens Kit for iPhone, Samsung, Pixel, Macro and Wide Angle Lens with LED Light and Travel Case: Camera & Photo. Retrieved May 2, 2019 from https://www.amazon.com/gp/product/B01A6D2JVI/ref=ppx_yo_dt_b_asin_title_o07_s00?ie=UTF8&psc=1

[3] Apple. SFSpeechRecognizer - Speech | Apple Developer Documentation. Retrieved September 18, 2018 from https://developer.apple.com/documentation/speech/sfspeechrecognizer?language=objc

[4] Apple. AVSpeechSynthesizer - AVFoundation | Apple Developer Documentation. Retrieved September 18, 2018 from https://developer.apple.com/documentation/avfoundation/avspeechsynthesizer?language=objc

[5] Blender. Blender. Retrieved April 25, 2019 from https://www.blender.org/

[6] 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.

[7] Gary R. Bradski and Adrian. Kaehler. 2008. Learning OpenCV : computer vision with the OpenCV library. O'Reilly.

[8] Anke M. Brock. 2013. Touch the map!: designing interactive maps for visually impaired people. ACM SIGACCESS Accessibility and Computing, 105: 9–14. https://doi.org/10.1145/2444800.2444802

[9] Anke M. Brock, Philippe Truillet, Bernard Oriola, Delphine Picard, and Christophe Jouffrais. 2015. Interactivity Improves Usability of Geographic Maps for Visually Impaired People. Human–Computer Interaction 30, 2: 156–194. https://doi.org/10.1080/07370024.2014.924412

[10] Anke Brock, Philippe Truillet, Bernard Oriola, and Christophe Jouffrais. 2011. Usage of multimodal maps for blind people. In ACM International Conference on Interactive Tabletops and Surfaces (ITS '10), 247. https://doi.org/10.1145/1936652.1936699

[11] Anke Brock, Philippe Truillet, Bernard Oriola, Delphine Picard, and Christophe Jouffrais. 2012. Design and user satisfaction of interactive maps for visually impaired people. In Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 544–551. https://doi.org/10.1007/978-3-642-31534-3_80

[12] John Brooke. 1996. SUS-A quick and dirty usability scale. Usability evaluation in industry 189, 194: 4–7.

[13] 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

[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, 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

[16] Julie Ducasse, Anke M Brock, and Christophe Jouffrais. 2018. Accessible Interactive Maps for Visually Impaired Users. In Mobility of Visually Impaired People. Springer, Cham, 537–584. https://doi.org/https://doi.org/10.1007/978-3-319-54446-5_17

[17] Stéphanie Giraud, Anke M. Brock, Marc J.-M. Macé, Christophe Jouffrais, Mohamed Chetouani, Marco Fyfe Pietro Gillies, M-Carmen Juan, Christophe Jouffrais, Stéphanie Giraud, Anke M. Brock, and Marc J-M Macé. 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. https://doi.org/10.3389/fpsyg.2017.00930

[18] Stéphanie Giraud, Philippe Truillet, Véronique Gaildrat, and Christophe Jouffrais. 2017. "DIY" prototyping of teaching materials for visually impaired children: Usage and satisfaction of professionals. In Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 515–524. https://doi.org/10.1007/978-3-319-58706-6_42

[19] 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

[20] Timo Götzelmann. 2018. Visually Augmented Audio-Tactile Graphics for Visually Impaired People. ACM Transactions on Accessible Computing 11, 8: 31. https://doi.org/10.1145/3186894

[21] Timo Götzelmann and Aleksander Pavkovic. 2014. Towards automatically generated tactile detail maps by 3D printers for blind persons. In International Conference on Computers for Handicapped Persons, 1–7. https://doi.org/10.1007/978-3-319-08599-9_1

[22] Noreen Grice, Carol Christian, Antonella Nota, and Perry Greenfield. 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. https://doi.org/10.5241/5-66

[23] Jaume Gual, Marina Puyuelo, and Joaquim Lloveras. 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, 155–164. Retrieved from http://www.scopus.com/inward/record.url?eid=2-s2.0-84858852744&partnerID=tZOtx3y1

[24] Jaume Gual, Marina Puyuelo, and Joaquim Lloveras. 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. https://doi.org/10.1177/0264619614540291

[25] Jaume Gual, Marina Puyuelo, Joaquim Lloveras, and Lola Merino. 2012. Visual Impairment and urban orientation. Pilot study with tactile maps produced through 3D Printing. Psyecology 3, 2: 239–250. https://doi.org/10.1174/217119712800337747

[26] 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

[27] Leona Holloway, Kim Marriott, and Matthew Butler. 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), Paper 198, 13 pages. https://doi.org/10.1145/3173574.3173772

[28] Ke Huo, Vinayak, and Karthik Ramani. 2017. Window-Shaping: 3D Design Ideation by Creating on, Borrowing from, and Looking at the Physical World. In Proceedings of the Eleventh International Conference on Tangible, Embedded, and Embodied Interaction (TEI '17)., 37–45. https://doi.org/10.1145/3024969.3024995

[29] R Dan Jacobson. Navigating maps with little or no sight: An audio-tactile approach. Retrieved April 25, 2019 from https://www.aclweb.org/anthology/W98-0214

[30] 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

[31] Jeeeun Kim and Tom Yeh. 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), 2815–2824. https://doi.org/10.1145/2702123.2702144

[32] Michael A. Kolitsky. Making Tactile 3D Prints Talk. Retrieved September 20, 2018 from http://www.instructables.com/id/Making-Tactile-3D-Prints-Talk/?ALLSTEPS

[33] Michael A. Kolitsky. 2014. 3D Printed Tactile Learning Objects: Proof of Concept. Journal of Blindness Innovation and Research 4, 1: 4–51.

[34] A.F. Koschan. Perception-based 3D triangle mesh segmentation using fast marching watersheds. In 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003. Proceedings., II-27-II–32. https://doi.org/10.1109/CVPR.2003.1211448

[35] Louisheath. Braille Printer. Retrieved from https://github.com/louisheath/braille-printer

[36] OpenSCAD. The Programmers Solid 3D CAD Modeller. Retrieved September 19, 2019 from https://www.openscad.org/

[37] OpenStreetMap. HaptoRender. Retrieved April 25, 2019 from https://wiki.openstreetmap.org/wiki/HaptoRender

[38] OpenStreetMap. OpenStreetMap. Retrieved from https://www.openstreetmap.org/

[39] Huaishu Peng, Jimmy Briggs, Cheng-Yao Wang, Kevin Guo, Joseph Kider, Stefanie Mueller, Patrick Baudisch, and François Guimbretière. 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), 1–12. https://doi.org/10.1145/3173574.3174153

[40] Hanae Rateau, Yosra Rekik, Edward Lank, and Laurent Grisoni. 2018. Ether-Toolbars: Evaluating Off-Screen Toolbars for Mobile Interaction. In 23rd International Conference on Intelligent User Interfaces (IUI '18), 487–495. https://doi.org/10.1145/3172944.3172978

[41] 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

[42] Rhino. Rhino 6. Retrieved April 28, 2019 from https://www.rhino3d.com/

[43] Jonathan Rowell and Simon Ungar. 2003. The world of touch: an international survey of tactile maps. Part 1: production. British Journal of Visual Impairment 21, 3: 98–104. https://doi.org/10.1177/026461960302100303

[44] Johnny Saldana. 2015. The Coding Manual for Qualitative Researchers. Sage.

[45] 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. Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology (UIST '15): 253–258. https://doi.org/10.1145/2807442.2807503

[46] 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

[47] Linda Sheppard and Frances K. Aldrich. 2001. Tactile graphics in school education: perspectives from teachers. British Journal of Visual Impairment 19, 3: 93–97. https://doi.org/10.1177/026461960101900303

[48] 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

[49] Lei Shi, Holly Lawson, Zhuohao Zhang, and Shiri Azenkot. 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), 14. https://doi.org/https://doi.org/10.1145/3290605.330042

[50] 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

[51] 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

[52] Lei Shi, Yuhang Zhao, and Shiri Azenkot. 2017. Markit and Talkit: A Low-Barrier Toolkit to Augment 3D Printed Models with Audio Annotations. In Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology (UIST '17), 493–506. https://doi.org/10.1145/3126594.3126650

[53] Alexa F. Siu, Son Kim, Joshua A. Miele, and Sean Follmer. 2019. shapeCAD: An Accessible 3D Modeling Workflow for the Blind and Visually-Impaired Via 2.5D Shape Displays. In The 21st International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS '19), 342–354. https://doi.org/10.1145/3308561.3353782

[54] SketchUp. SketchUp. Retrieved April 28, 2019 from https://www.sketchup.com/

[55] Abigale Stangl, Chia-Lo Hsu, and Tom Yeh. 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), 127–137. https://doi.org/10.1145/2700648.2809854

[56] 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

[57] Saiganesh Swaminathan, Thijs Roumen, Robert Kovacs, David Stangl, Stefanie Mueller, and Patrick Baudisch. 2017. Linespace: A sensemaking platform for the blind. Informatik-Spektrum 40, 6: 516–526. https://doi.org/10.1007/s00287-017-1070-1

[58] TactileView. TactileView. Retrieved April 28, 2019 from http://www.tactileview.com/

[59] 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

[60] Brandon T. Taylor, Anind K. Dey, Dan P. Siewiorek, and Asim Smailagic. 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), 427–428. https://doi.org/10.1145/2700648.2811336

[61] Thingiverse. Digital Designs for Physical Objects. Retrieved April 28, 2019 from https://www.thingiverse.com/

[62] Tinkercad. Create 3D digital designs with online CAD. Retrieved April 28, 2019 from https://www.tinkercad.com/

[63] Touch Mapper. Tactile Maps Easily. Retrieved April 25, 2019 from https://touch-mapper.org/en/

[64] Vvoovv. Blender OSM. Retrieved from https://github.com/vvoovv/blender-osm

[65] Henry B. Wedler, Sarah R. Cohen, Rebecca L. Davis, Jason G. Harrison, Matthew R. Siebert, Dan Willenbring, Christian S. Hamann, Jared T. Shaw, and Dean J. Tantillo. 2012. Applied Computational Chemistry for the Blind and Visually Impaired. Journal of Chemical Education 89, 11: 1400–1404. https://doi.org/10.1021/ed3000364

[66] Christian Weichel, Manfred Lau, David Kim, Nicolas Villar, Hans W. Gellersen, Christian Weichel, Manfred Lau, David Kim, Nicolas Villar, and Hans W. Gellersen. 2014. MixFab: a mixed-reality environment for personal fabrication. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '14), 3855–3864. https://doi.org/10.1145/2556288.2557090