A Demo of Talkit++: Interacting with 3D Printed Models Using iOS Devices

Lei Shi
Information Science,
Cornell Tech
New York, NY U.S.A.
ls776@cornell.edu

Zhuohao Zhang
Computer Science,
Zhejiang University
Hangzhou, China
zhuohaozhang@zju.edu.cn

Shiri Azenkot
Jacobs Technion-Cornell
Institute, Cornell Tech
New York, NY U.S.A.
shiri.azenkot@cornell.edu

ABSTRACT
Tactile models are important learning materials for visually impaired students. With the adoption of 3D printing technologies, visually impaired students and teachers will have more access to 3D printed tactile models. We designed Talkit++, an iOS application that plays audio and visual content as a user touches parts of a 3D print. With Talkit++, a visually impaired student can explore a printed model tactilely and use hand gestures and speech commands to get more information about certain elements in the model. Talkit++ detects the model and hand gestures using computer vision algorithms, simple accessories like paper stickers and printable trackers, and the built-in RGB camera on an iOS device. Based on the model's position and the user's input, Talkit++ speaks textual information, plays audio recordings, and displays visual animations.

Author Keywords
Computer vision; 3D printing; visual impairments

ACM Classification Keywords
H.5.m. Information interfaces and presentation (e.g., HCI)

INTRODUCTION
Teachers for the visually impaired (TVIs) often use tactile materials (including 3D models) when teaching students with visual impairments a variety of concepts. In prior work, we presented Talkit [6, 7], an application that adds audio labels to 3D printed models. Talkit aims to make tactile materials more powerful by making them interactive. As 3D printers become more popular, our goal is to enable TVIs to create custom interactive tactile materials with which they can teach their students more effectively.

Talkit was a laptop application that augmented areas of a 3D printed model with audio labels. When a user tactilely explored these areas, Talkit spoke their labels. To enable Talkit, we designed Markit, a design tool with which people can associate areas of a model with audio labels [7].

Figure 1. We designed three interactive models: (a) the Map model, (b) the Volcano model, and (c) the Plane model.

In this demo, we present Talkit++, a new version of Talkit. Through two design workshops with teachers of the visually impaired (TVIs), we identified several limitations of the Talkit application, and developed Talkit++ to address these limitations. After the workshops, we worked with three TVIs individually to design and implement interactive printed models in their classrooms, as shown in Figure 1.

Compared to Talkit, Talkit++ supports multiple types of output information, scalable 2D trackers (used for tracking the models with computer vision), and iOS compatibility. Besides descriptive textual information (e.g., the name of a bird), Talkit++ also supports audio recordings (e.g., the sound a bird makes) and visual animations (e.g., the animation of a flying bird). In addition to 3D trackers, Talkit++ also supports 2D trackers, and the sizes of the trackers can be changed, as shown in Figure 1a and 1b. Talkit++ is implemented on iOS, which will allow us to more easily deploy and evaluate the system in the wild. While this demo focuses on Talkit++, we also updated Markit, the model design tool, to support the new features in Talkit++.

EVALUATING TALKIT
We iterated on the design of Talkit based on participatory design workshops we conducted with TVIs. These workshops were conducted in two regional conferences of the Association for Education and Rehabilitation of the Blind and Visually Impaired (AERBVI), an organization for professionals who provide services to visually impaired people [2]. A total of 37 TVIs participated in the workshops.

Each workshop lasted 1.5 hours. We began with a 15-minute presentation about Talkit, Markit, and a small set of sample interactive models. We then split the participants

into groups with 3 to 5 members, and asked each group to choose a 3D printed model and design interactive elements for the model. Meanwhile, they tried out the Talkit application with existing interactive models. In the end, each group presented the model they designed along with features they would like to see incorporated into Talkit.

We summarize key takeaways from their feedback:

1. Overall, the TVIs were interested in using Talkit and interactive models with their students.
2. The Talkit application only supported descriptive textual information. However, audio recordings and visual content (for low vision students) were also important.
3. Talkit only supported 3D trackers, which could be bulky on some models.
4. Talkit ran on a laptop using its built-in camera. It's difficult to map the visual field of the camera to one's own visual field.

To improve the system, we incorporated new features. In addition, we implemented the system on iOS devices.

TALKIT++

Like Talkit, Talkit++ uses printable trackers and a paper sticker. Talkit++ locates a model by tracking the trackers, and recognizes finger gestures using the paper sticker.

To create an interactive model, a model designer uses the updated Markit application to add trackers and output information. The trackers enable Talkit++ to locate the model from the video stream. A designer may choose the types of trackers for different models. For example, the designer may use a 3D tracker on a plane model, since a visually impaired student may rotate the plane model during exploration. For models that involve less spatial movement (e.g., the Map model in Figure 1), the designer can use 2D trackers.

In addition to the trackers, the designer needs to associate the elements on the model with output information. The designer may choose different types of output information. For example, she could associate an element with descriptive texts (e.g., "engine"), the files of audio recordings (e.g., the sound of an engine), the files of visual animations (e.g., the internal movement of an engine).

To use an interactive model, a visually impaired user sticks a red sticker on his fingernail, and runs Talkit++ on his iOS device (e.g., an iPhone or an iPad). Figure 2a shows the user interface of the application, while Figure 2b illustrates one setup a student used.

IMPLEMENTATION

Talkit++ has three main functionalities: (1) locating the model, (2) detecting the user's finger gesture, and (3) recognizing speech commands. For detailed descriptions about the algorithms of Talkit++, please refer to our previous work [7]. We briefly describe the implementation of the system in this section.

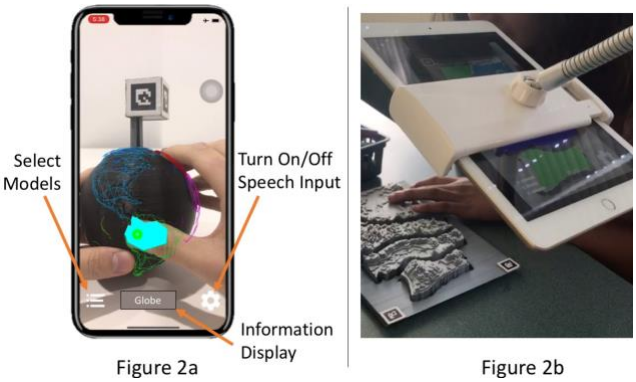

Figure 2. (a) The user interface of the Talkit++ application. (b) A visually impaired student is using Talkit++ on an iPad to learn the Map model.

Talkit++ was developed in Swift and C++. To locate a model, Talkit++ uses Chilitags [3], which provides the coordinates of 2D and 3D trackers. Based on these coordinates, the system calculates the real-world coordinates of the model, and tracks a user's red sticker using OpenCV [4]. We cross compiled Chilitags and OpenCV and called them in Swift. After locating the model, Talkit++ provides output information based on the location of the sticker. As for speech input, Talkit++ uses Swift's built-in speech dictation functions.

SAMPLE MODELS

After developing Talkit++, we co-designed interactive models with three TVIs to design models for their students. We held weekly remote meetings with each TVI. During these meetings, the TVIs provided ideas and teaching plans, and the researchers helped them design the models. The entire design activity took one month, and produced three interactive models, as shown in Figure 1:

1. *Map*: A Washington State Regions map modified from a topographical map [8]. A TVI annotated the five regions on the map with labels, descriptions, and audio recordings. For example, she added the sound of an ocean for the "Coastal Range" region.
2. *Volcano*: A volcano model modified from Thing: 1316498 [1]. The model had five annotated elements. In addition to audio information, a TVI designed an animation of a volcano eruption for her low vision students.
3. *Plane*: A plane model modified from Thing: 182252 [5]. The model had six annotated elements, and a TVI added audio recordings for the cockpit, the body, and the jet engines of the plane model.

FUTURE WORK

In the future, we plan to make the application available to the public. In addition, we will create an online repository for interactive 3D printed models. The repository will collect practical samples of interactive models, and enables teachers and students to access these models freely.

REFERENCES

[1] 3D Print a Volcano by mfritz - Thingiverse: *https://www.thingiverse.com/thing:1316498*. Accessed: 2018-06-17.

[2] AER – Association for Education and Rehabilitation of the Blind and Visually Impaired: *https://aerbvi.org/*. Accessed: 2018-06-17.

[3] Bonnard, Q. et al. 2013. Chilitags 2: Robust Fiducial Markers for Augmented Reality and Robotics. CHILI, EPFL, Switzerland.

[4] Bradski, G.R. and Kaehler, A. 2008. *Learning OpenCV : computer vision with the OpenCV library*. O'Reilly.

[5] Plane by YahooJAPAN - Thingiverse: *https://www.thingiverse.com/thing:182252*. Accessed: 2018-06-17.

[6] Shi, L. et al. 2016. Magic Touch: Interacting with 3D Printed Graphics. *Proceedings of the 18th International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS '16)* (2016), 329–330.

[7] Shi, L. et al. 2017. Markit and Talkit: A Low-Barrier Toolkit to Augment 3D Printed Models with Audio Annotations. *UIST '17* (2017).

[8] Washington State Topographical Map | 3D CAD Model Library | GrabCAD: *https://grabcad.com/library/washington-state-topographical-map*. Accessed: 2018-06-17.