Facade: Auto-generating Tactile Interfaces to Appliances

Anhong Guo[1], Jeeeun Kim[2], Xiang 'Anthony' Chen[1], Tom Yeh[2], Scott E. Hudson[1], Jennifer Mankoff[1], Jeffrey P. Bigham[1]

[1] Human-Computer Interaction Institute, Carnegie Mellon University, Pittsburgh, PA, USA
[2] Department of Computer Science, University of Colorado Boulder, Boulder, CO, USA

{anhongg, xiangche, scott.hudson, jmankoff, jbigham}@cs.cmu.edu,
{jeeeun.kim, tom.yeh}@colorado.edu

Figure 1. Facade is a crowdsourced fabrication pipeline that enables blind people to make flat physical interfaces accessible by independently producing a 3D-printed overlay of tactile buttons. From left to right, we demonstrate example applications including microwave, refrigerator door, copier, and another microwave. Insets shows close views of individual embossed buttons.

ABSTRACT

Common appliances have shifted toward flat interface panels, making them inaccessible to blind people. Although blind people can label appliances with Braille stickers, doing so generally requires sighted assistance to identify the original functions and apply the labels. We introduce *Facade*—a crowdsourced fabrication pipeline to help blind people independently make physical interfaces accessible by adding a 3D printed augmentation of tactile buttons overlaying the original panel. Facade users capture a photo of the appliance with a readily available fiducial marker (a dollar bill) for recovering size information. This image is sent to multiple crowd workers, who work in parallel to quickly label and describe elements of the interface. Facade then generates a 3D model for a layer of tactile and pressable buttons that fits over the original controls. Finally, a home 3D printer or commercial service fabricates the layer, which is then aligned and attached to the interface by the blind person. We demonstrate the viability of Facade in a study with 11 blind participants.

ACM Classification Keywords

H.5.2 Information interfaces and presentation: User Interfaces - *Input devices and strategies*; K.4.2 Computers and Society: Social Issues - *Assistive technologies*

CHI 2017, May 06-11, 2017, Denver, CO, USA

DOI: http://dx.doi.org/10.1145/3025453.3025845

Author Keywords

Non-visual interfaces; visually impaired; blind; accessibility; crowdsourcing; fabrication; 3D printing; computer vision.

INTRODUCTION

Flat touchpads have proliferated on common appliances, making them inaccessible for blind people. The task of creating an appropriate tactile overlay to adapt to inaccessible appliances currently requires in-person sighted help and a labeling device that can print embossed labels. However, sighted assistance is not always available, and a labeling device doesn't solve issues such as layout and size of labels. Automatically generated tactile overlays could address both issues. We present an end-to-end crowdsourced fabrication pipeline that can be done independent of in-person sighted help, and costs less than $10 per appliance.

To identify the existing challenges of using inaccessible interfaces of home and work appliances, we conducted a formative study with six blind participants. We identified four design requirements for a system to augment physical interfaces for non-visual access: *(i)* the solution for tactile labeling should enable blind users to independently augment and access their appliances without in-person sighted assistance, *(ii)* the augmented labels should be customizable to address individual needs, *(iii)* the solution should allow for learning and memorization of the interface, and *(iv)* the tactile labels should support easy attachment and reproduction for repeated use.

We introduce *Facade*, a crowdsourced fabrication pipeline to make physical interfaces accessible by adding a 3D printed layer of tactile buttons overlaying the original panel (Figure 2). When a blind person encounters an inaccessible appliance for the first time, s/he uses the Facade iOS app to capture a

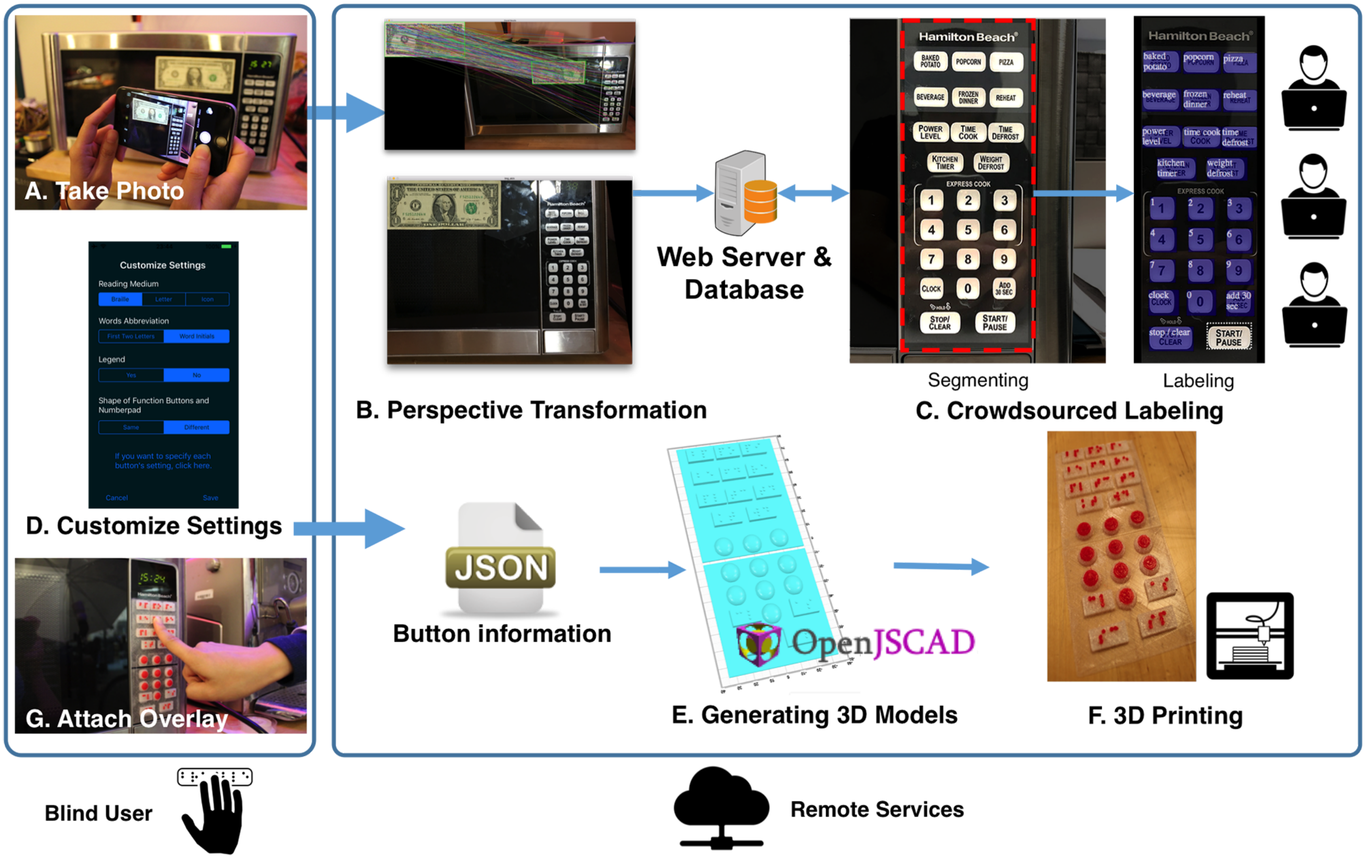

Figure 2. Facade users capture a photo of an interface they would like to use with a fiducial marker attached to it (we use a dollar bill). Using perspective transformation, the interface image is warped to the front view and absolute measurements are calculated. Then this image is sent to multiple crowd workers, who work in parallel to quickly label and describe elements of the interface. Blind users can then customize settings of the labeling strategy, and these labels and preferences are used to generate the 3D models of a tactile layer matching the original controls. Finally, an off-the-shelf 3D printer fabricates the layer, which is then attached to the interface using adhesives.

photo of the interface using a dollar bill as a fiducial marker for recovering size information (Figure 2A and B). Within a few minutes, crowd workers mark the layout of the interface, annotate its elements (*e.g.*, buttons or other controls), and describe each element (Figure 2C). These labels are then used to generate 3D models of a layer of tactile and pressable buttons matching the original controls (Figure 2E), which the blind users can customize by changing the shape and labels of the buttons using the Facade iOS app (Figure 2D). Finally, an off-the-shelf 3D printer can be used to fabricate the layer (Figure 2F). The printed button facade is designed to be easily aligned and attached to its appliance using adhesives (Figure 2G). Although consumer-grade 3D printers might not be readily available to blind people at home, many printing services are available from which a print can be mail-ordered. In addition, we can expect that consumer-grade printers will continue to improve in speed and robustness. Even with mail-order costs, Facade is an inexpensive ($10 from a service such as 3D Hubs[1]) and more accessible alternative solution.

[1] https://www.3dhubs.com

This paper makes the following contributions:

- In a user study, we identify existing challenges and design requirements for augmenting physical interfaces with tactile markers.
- We introduce Facade, a crowdsourcing and fabrication pipeline to augment inaccessible physical interfaces with overlaid 3D printed tactile buttons.
- Our validation shows that Facade enables blind people to independently augment appliance interfaces, and that fabricated overlays provide rich and usable tactile feedback for accessing otherwise inaccessible appliances.

RELATED WORK

Recent advances in consumer-grade 3D printers and the do-it-yourself (DIY) movement have changed the audience of 3D printing. It has already been established as a tool that enables amateurs to create a wide variety of assistive technologies [11, 12, 24, 31]. However, the barriers to entry for 3D modeling custom assistive technologies are high, which has lead to research on tools that can support amateurs without requiring mastery of modeling (*e.g.,* [16]). In addition, assistive technology must typically interoperate with existing objects in the real world, which brings new challenges such as attachment [15] and interoperation [36].

In terms of accessibility for blind users, 3D printing has been used to produce custom labels on 3D printed objects [37], generate tactile maps [10, 19, 20, 39], support literacy skills through the creation of tactile picture books [27], teach design [32], mathematics [13], programming [26] and deliver tactile visualizations [9, 38]. These applications of 3D printing share a focus on 3D representations that can be customized beyond what is possible with the current state of the art (thermal printing or Braille labeling). However, this body of work assumes a sighted person who designs and produces the 3D printed artifact, which may limit a blind person's ability to access 3D printed solutions as needed.

One potential way of reducing the barriers to accessing sighted assistance is to shift the work to a virtual crowd [3, 6]. A number of crowd-powered systems have been developed to make visual information accessible to blind people [7]. VizWiz lets blind people take a picture, speak a question, and get answers back from the crowd within approximately 30 seconds [4]. Chorus:View [29] pairs a user with a group of crowd workers using a shared video stream. "Be My Eyes"[2] matches users to a single volunteer over a shared video stream. VizWiz::LocateIt [5] allows blind people to ask for assistance in finding a specific object. RegionSpeak [46] enables spatial exploration of the layout of objects in a photograph using a touchscreen. VizLens [21, 22] fuses crowdsourcing and computer vision to robustly and interactively help blind people use inaccessible interfaces in the real world, similar to a screen reader. Recently, physical crowds have been organized to construct pre-designed large-scale structures [28]. However, crowds have not in the past been used to create custom 3D printed objects. Facade combines a crowd interpretation pipeline with an accessible 3D printing application [23].

Another approach is to create new devices that are accessible, but this is unlikely to make all devices accessible due to cost. As more and more devices are connected to the Internet and can be controlled remotely, the problem becomes one of digital accessibility, which is easier to solve. For example, users may bring their own smartphone with an interface that is accessible to them, and use it to connect to the device [17, 35, 41]. Facade handles the legacy of inaccessible devices, which neither approach does.

To summarize, 3D printing can produce customized physical augmentations, and crowdsourcing can release the constraints of in-person sighted help through online and always-available visual assistance. Both have been applied with success in the domain of accessibility, including addressing the needs of blind users. Facade's novel contribution is in bringing these threads of research together to solve the important problem of making everyday appliances accessible.

FORMATIVE STUDY

To better understand how blind people currently use and accommodate home and office appliances, we conducted a formative study with 6 blind participants (all female, age 34-73). Four of the participants were congenitally blind, and the other two had light perception. All were Braille readers.

Procedure

We first went to the home of a blind individual, and observed how she cooked a meal and used home appliances. We then conducted semi-structured interviews with all participants. We asked questions about home appliance use, whether these appliances were accessible, if not, the ways employed to use these appliances, and strategies to label them. The studies were documented with video and audio recordings, as well as handwritten notes. We extracted key quotes and themes that reflected participants' personal strategies and challenges.

Results: Design Considerations

Participants remarked that interfaces are becoming much less accessible as flat digital touch pads replace physical buttons, which can at least be easily found by fingers once the locations of different functions were memorized. Appliances mentioned by participants were very diverse, and their interfaces differed in size, label, type of functions and number of buttons.

We identified four design requirements for a system to generate augmented physical interfaces for non-visual access. We refer to the participants in our formative study as F1 - F6 below (and also include related comments gathered later from our evaluation study participants P1 - P11).

Independence

Blind users often depend on in-person sighted assistance to identify the original functions and apply the labels on home appliances. When they bought a new appliance, they needed to wait for sighted help before being able to use the appliance.

> *My brother let me and my husband know what buttons are, we decide what buttons matter for us. And we write the Braille to label them, he again help us to stick onto buttons.* (P2)

The problem of existing solutions of applying Braille stickers, is that blind people cannot independently make appliances accessible. To address this, our solution should enable blind users to independently augment appliance interfaces, without needing to wait for help from sighted people.

Custom Settings

Participants had their own preferences and strategies for labeling. Simple dots (which could easily be counted and felt at a glance) were a popular choice on number buttons. Although not identical to Braille numeric characters, Braille readers also liked this strategy and only used Braille labels on more complex features, such as soft/melt, cook time, reheat, defrost, cancel and start on the microwave.

However, participants said they don't need all the buttons to be labeled. Some of them put bump dots or easily recognizable marker on frequently used buttons.

> *I put bump dots on only the 'add 30 seconds' button that I frequently use.* (F1)

When all the buttons are labeled with the same Braille dots, it's harder for them to find the number pad. Some of them mark only one of the number buttons (*e.g.,* 0 or 5) as the reference to identify all others. F5 suggested that differentiating the number pad from other buttons could make interacting with the microwave faster.

[2] http://www.bemyeyes.org

Please indicate where the number starts, and that is enough. I can identify where other buttons are, it will make tasks quicker. (F2)

I do not need to mark the entire number pad. 0, left and right are enough to get where number buttons are. (F3)

Our solution should accommodate different preferred labeling strategies and reading mediums (Braille, printed letters, or dots). It should also use different shapes for functional buttons and number pads to reduce searching time.

Memorization Strategy

Since blind people were not familiar with the appliance functions, when using in-person sighted help for identifying the original functions and applying the labels, it was hard for them to remember the abbreviations and functions for more than a few buttons [33]. Therefore, they only tended to label a few buttons with only one or two Braille letters due to the limited size of the buttons, which limited their access to the appliances. There are also appliance interfaces that are hard to label. F1 reported making legend for a toaster oven since the buttons are hard to add labels on. Related to it, P5 stated:

I have an index card for a washer in my apartment, what normal hot and normal warm buttons are. I had my mom to help me to label when I moved in long time ago. (P5)

To address this, our solution should better support learning and memorization of the appliance functions through the use of in-app support, or physical legend.

Robustness

The Braille labels applied to the interface will wear out over time. When it happened, blind people lost access to the specific buttons, and required sighted help again to reapply the labels.

We use microwave in the kitchen with dirty hands. Braille stickers are so easily fall off. (P2)

To solve this problem, our solution should allow blind users to easily do the attachment independently. Furthermore, it should support easy reproduction and decrease the amount of effort required for the repeated work on the same appliance.

To summarize, our findings indicate that a solution for tactile labeling should allow blind users to independently augment and access their appliances. The solution should also support rich tactile feedback, diverse labeling strategies and preferences to address a wide range of individual needs. Furthermore, the solution should allow for learning and memorization of the interface, as well as easy attachment and reproduction.

FACADE

Assisted by the Facade iOS app, blind users capture a photo of an inaccessible interface with a readily available fiducial marker (a dollar bill) for recovering size information. The web server transforms the image to the front perspective then feeds this image to multiple crowd workers, who work in parallel to quickly label and describe elements of the interface. These labels are then used to generate a 3D model for a layer of tactile and pressable buttons matching the original controls, which blind users can customize by changing the shape and labels of the buttons using the Facade iOS app. Finally, a home 3D printer or service fabricates the layer, which is then aligned and attached to the interface by blind users. Facade works as a pipeline, and is fully automated. Users do not need to attend to its full complexity.

Capture and Perspective Transformation

The first time a user encounters an interface, s/he uses the Facade iOS app to take a photo of the interface with a dollar bill (Figure 2A), and sends the image to be processed and pushed to the crowd for manual labeling. The dollar bill is used to produced an image of the interface warped to appear as if from the front perspective, and to recover size information. We use a dollar bill as an example to demonstrate the utility of using currency bills as fiducial markers because of its ubiquity, its standard size and appearance, and its richness in details and texture to provide sufficient feature points for tracking. We expect that a deployed version of Facade would allow users to choose their preferred bill in their local currency.

Facade uses SURF (Speeded-Up Robust Features) [2] feature detector to compute key points and feature vectors in both the standard image of the dollar bill and the input image. Then the feature vectors are matched using FLANN (Fast Library for Approximate Nearest Neighbors) [34] based matcher. By filtering matches and finding the perspective transformation [44] between the two images using RANSAC (Random Sample Consensus) [18], our system is able to localize the standard dollar bill image in the input image, and warp the input image to the front perspective for further labeling. Figure 2B shows the results of perspective transformation using a dollar bill. Using a system similar to VizLens [22], the Facade app streams images to the backend server, which then localizes either side of the dollar bill in the image and provides real-time feedback on the aiming of camera relative to the dollar bill to blind users. By reading out instructions such as “not found”, “move phone to left/right/up/down/further” and “aiming is good”, the app guides the blind user to more easily take a photo from the front perspective, which will result in better warped image after the perspective transformation. The computer vision components are implemented using C++ and the OpenCV Library.

Facade only has knowledge of the dollar bill and provides guidance based on its location, without knowing where the interface is. Blind users use this guidance, combined with their knowledge of the relative location of the interface and the dollar bill, to aim the camera and take photos. However, if the appliance interface is partially cropped in the photo, in the next step, crowd workers will provide feedback to the user for taking another photo. Using a second marker could help, but appliances might not have enough space to fit two markers. In the future, we could use more advanced techniques for helping blind users take photos [25, 30, 42, 43, 45].

Crowdsourced Segmenting and Labeling

Facade uses a two-step workflow to label the area of the image that contains the interface and then label the individual visual elements (Figure 2C), similar to those in VizLens [22]. Crowd workers are first asked to rate the image quality, and segment the interface region. Results are combined using majority vote. To assist with later attachment, we ask crowd workers to segment the interface region aligned with the physical boundaries of the appliance interface, so that blind people can feel that boundary and align the overlay themselves.

Crowd workers are then instructed to draw bounding boxes around all of the individual buttons within the interface area, and provide a text annotation for each element (such as labeling buttons as ‘baked potato’, ‘start/pause’). In this step, crowd workers work in parallel, and the worker interface shows labeled elements to other workers as they are completed.

Fabricating Accessible Augmented Layer

Labels are used to generate a 3D model for a tactile and pressable button layer, matching the original controls. After labeling by crowd workers, the blind user can use VoiceOver to customize the preferences for the tactile layer to be fabricated using the iOS app (Figure 2D). Blind users specify customizations using a virtual version of the interface displayed on their iPhone. Informed by our study, we allow individual buttons to be customized using Braille, embossed letters, or embossed symbols. Although embossed capital letters were not mentioned in our study, blind participants did mention using shared machines at home and at work with sighted people, which embossed letters allows for co-located access. Embossed letters also improve access for non-Braille readers, who can recognize capital letters almost as well as Braille readers recognize Braille [14]. Finally, users can customize the abbreviation strategy (*i.e.*, which letters are used to represent a word or phrase); request a legend; edit the tactile label of individual buttons; set which buttons to label or remain flat; and customize the shape of buttons (useful for differentiating special buttons such as numbers).

Based on the results from our formative studies, we decide by default to detect and use different shapes for function (rectangular) and number (spherical) buttons when generating the 3D tactile overlay. Following common numeric keyboard or button pad accessibility conventions [40], by default we only label number 5 with a dot on the spherical button for the numbers.

The settings and the crowd-generated labels are then passed to our automated design tool. We implemented an OpenJSCAD script to generate the final STL files of 3D models of the augmented buttons for printing (Figure 2E). The input to the program is a generated JSON object including the dimensions of the tactile overlay, average button size, as well as the dimensions, positions, labels and preferences of each button. With this data, the script first generates groups of labeled 3D buttons. We determine the depth of the buttons to be proportional to the size of the buttons. To get the scale factor for Braille and letters, we first divide the button width by two to situate two characters, and then divide each area to hold two columns and three rows of dots including spacing. Compared to the standard dot radius and spacing size [1], the proportion is defined by this scale factor, and applied to determine the size of letters and symbols.

If short acronyms are not provided for each button label, the program automatically generates the abbreviations. By default, when adding Braille on top of the buttons, we use two characters for each button due to the limited surface area and the size of Braille characters: a word (*e.g.*, ‘Clock’) is abbreviated by the first two letters (*e.g.*, ‘CL’); and multiple words (*e.g.*, ‘Power Level’) are abbreviated by the initial letters of the first two words (*e.g.*, ‘PL’). When requested, a separate

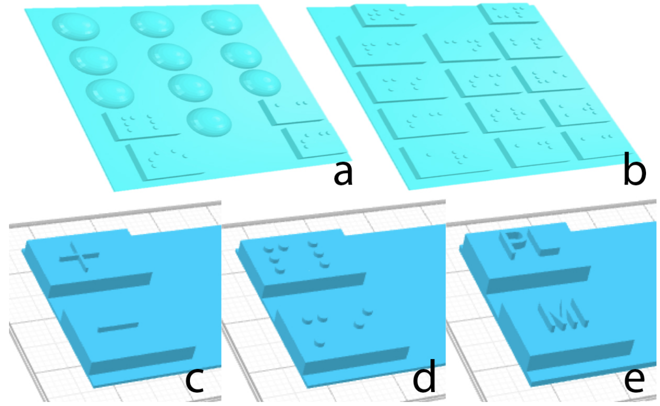

Figure 3. Shapes inform users of different functionalities. For example, half spherical buttons without Braille label indicates number buttons *(a)*, while rectangular buttons with Braille labels indicate function buttons *(b)*. Users are also able to change settings to use symbols *(c)*, Braille *(d)*, or embossed letters *(e)* for buttons labels such as plus and minus.

STL file is generated containing a legend (Figure 5e) detailing the abbreviations of the button labels, with the first column being acronyms, and the second column being the full words.

Our automated design tool then places buttons on top of a thin (2 layers in Gcode, 0.8mm) flat sheet, which creates a flat surface below the buttons that is easily attached to appliances with adhesives. Then, the program splits the tactile overlay into separate groups according to the 3D printer’s print bed size limit, and combines all sheets, buttons and embossed labels in each group into one piece for printing. The script can also merge multiple pieces as one print job based on print bed size to reduce print time. Our system exports files in ready-to-print STL format, which can be printed at the blind user’s home or through a commercial 3D printing service. An example 3D printed tactile overlay for a microwave is shown in Figure 2F. The overlay design in Figure 3 was finalized after several design iterations as we detail in the next section.

DESIGN ITERATIONS

To produce the most effectively functioning tactile overlay, we went through several design iterations. The microwave we chose as the testing device was a Hamilton Beach 1.1 Cu Ft Microwave (Figure 4c). Similar to most common microwaves, buttons on this microwave are flat and provide little (if any) tactile feedback. It contains some familiar buttons (0-9), and many that are likely to be less familiar (*e.g.*, time defrost, baked potato). All of our tactile overlays used in design iterations and user evaluations were produced with off-the-shelf consumer grade 3D printers using the FDM (fused deposition modeling) technique of 3D printing.

Iteration #1: Design Probe

To test the 3D printed Facade overlay, we first created a design probe—a 3D printed sheet in PLA plastic of buttons labeled with Braille acronyms, attached to the microwave (Figure 4). We used an inverted cone shape for buttons, with the radius of the top surface corresponding to the actual size of the original button, and the radius of the bottom surface smaller (Figure 4a). Thus, the design reduces the pressure required for blind users to press on the top surface to activate the original buttons on

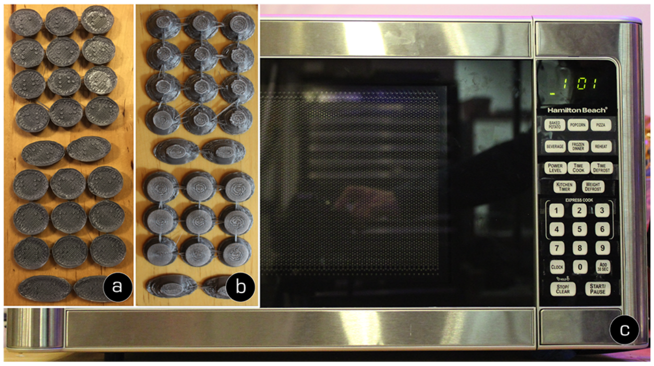

Figure 4. A design probe tested with 6 blind participants. An augmented button set with Braille labels *(a)* is attached to the microwave *(c)*, and buttons are connected with thin bridges to facilitate pressing *(b)*.

the microwave. To minimize assembly time, we attached the buttons in a grid with connectors between buttons (Figure 4b), so that they could be batch printed, and also attached to the physical interface as a whole. We also made the connectors very thin so that the plastic buttons deform more easily when pressed. All of this design work was done by hand, but in a style that can be automatically generated for Facade.

We tested this design with the same participants from our formative study, and identified the following issues:

- Some unexpected 3D printed artifacts on the edges of the top surface made the Braille dots feel overly rough, reducing legibility.
- Due to print resolution, Braille dots had different heights, reducing legibility.
- The plastic buttons were too hard to push.
- The button set did not attach to the microwave panel well and fell off after several times of use, due to the small contact regions.
- Because PLA does not deform, the connector bridges broke after pressing for a few times.

Iteration #2: Material Exploration

Informed by the participants' feedback to our initial design probe, we modified the design of the tactile overlay, and tested different combinations of materials (Figure 5a-d) to improve attacheability, legibility, and pressibility. Using a flat and thin sheet printed in flexible NinjaFlex[3] as the base of the overlay can make the augmentation much easier to attach to the appliance interface with adhesive.[4] The flexible material also made it much easier to press than using only PLA for the design probe.

While using NinjaFlex can improve attachability and pressability, it sometimes leaves undesired artifacts in the form of fine threads between Braille dots (think of melted cheese threads between pizza slices). These threads could reduce Braille legibility. One solution is to print Braille using hard material such as PLA (which we denote as Flex+PLA label), as shown in Figure 5b. A problem that occurred with this design is

Device	Material	Label
Hamilton Beach microwave	Flex*	Braille (Fig. 5a)
	Flex+PLA label	Braille (Fig. 5b)
	Flax+PLA cover	Braille (Fig. 5c)
	Flex+PLA cover	Letters (Fig. 5d)
	PLA legend	Braille (Fig. 5e)
Frigidaire Gallery microwave	Flex	Braille
Frigidaire fridge	Flex+PLA label	Braille
KitchenAid fridge	Flex	Letters + Symbol
Sharp microwave	Flex	Braille
Richo Alficio MP 6500 Copier**	Flex+PLA cover	Embossed letters Printed full words

Table 1. Interface, material and reading medium combinations used in design iteration 2 to improve attacheability, legibility, and pressibility.
*** Flex refers to NinjaFlex or SemiFlex for flexible material printing**
**** Required manual intervention for raised buttons**

that these Braille dots may become dislodged from the button surface over time, due to the combination of heterogeneous materials. Another solution for this is to print several layers of the button together with Braille dots in PLA, while printing the rest of the bottom layers in NinjaFlex, resulting in a larger contact area between the two materials to allow them to stick together nicely (which we denote as Flex+PLA cover). Table 1 summarizes our experiments on various printing mediums and material combinations for a wide variety of home appliances.

We then obtained formative feedback of the examples shown in Table 1 from one blind individual (female, 24 years old, college student). In three different settings (*i.e.*, pure NinjaFlex, Flex+PLA label, and Flex+PLA cover), the participant said all three testing material combinations were equally legible. Interestingly, she was most comfortable with reading the pure NinjaFlex version of the tactile overlay, despite the fine threads across dots. Unfortunately, both Flex+PLA label and Flex+PLA cover versions required her to press the button a lot harder to trigger the original interface. Overall, the NinjaFlex version of the tactile overlay had the best pressability and attachability among all material combinations we explored.

Iteration #3: Improved Legibility

Since the NinjaFlex version of printed Braille has enough detail and so is easily legible by a user, we printed the entire overlay in pure NinjaFlex including Braille. As guided by the user who tested our second design above, we also improved our design of the embossed letter version to make the letters thinner with larger gaps between letters for distinction.

For this improved design, we further tested the legibility of the Braille labels with two blind individuals (one female), both of whom provided formative feedback on the design. One suggested making the Braille dots more distinctive by raising the dots higher or reducing the button height. The other Braille expert, who works for a Braille publisher, suggested that Braille dots with a convex top are easier to read by touch than with a flat cylindrical top because convex tops provide a more salient separation between adjacent dots. Therefore, we changed the Braille dots from cylinders to domes, and finalized our design for the user evaluation we present next.

[3] https://ninjatek.com/products/filaments/ninjaflex/
[4] We used 3M removable double-sided Scotch Tape.

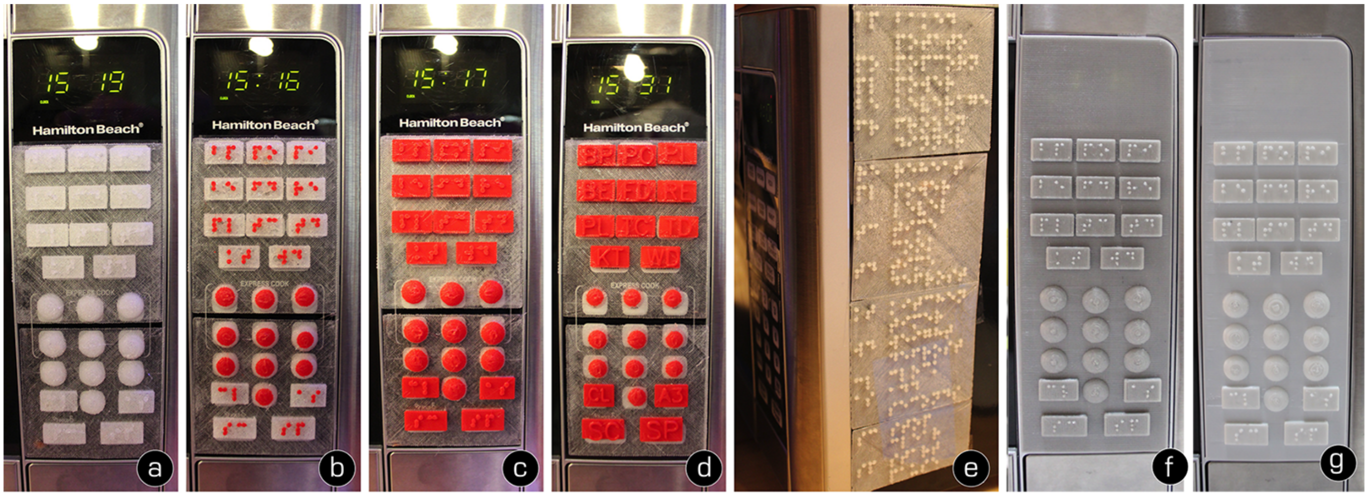

Figure 5. Example printed overlays and legends generated by Facade. ***(a)-(d)*** **demonstrate the different material combinations we tested in the design iterations (NinjaFlex with Braille, Flex+PLA Braille label, Flex+PLA Braille cover, and Flex+PLA embossed letter cover). Facade users can choose to print a legend for the abbreviations** ***(e)*****. If a user does not have a 3D printer at home, models can also be printed through commercial printing services and mail-ordered.** ***(f) and (g)*** **show two example prints ordered from 3D Hubs using PolyFlex and SemiFlex materials.**

USER EVALUATION

The goal of our user study was to evaluate whether Facade allows blind people to independently augment appliance interfaces, and how the fabricated overlay performs in assisting blind people accomplish realistic tasks that involve otherwise inaccessible interfaces. Our user evaluation included each step that the blind user needs to do in Facade.

Apparatus and Participants

We used the same inaccessible microwave as detailed in design iterations. The Facade iOS app was used in the study, installed on an iPhone 5c, runing iOS 9.3.4. For this particular evaluation, all the images were labeled by the experimenter as introducing the crowd would result in compounding factors. The tactile overlays used in the study were generated and 3D printed beforehand to save time. The quality of perspective transformation, crowdsourced labeling, and model production is presented in the next section ("Technical Evaluation"). We recruited 11 blind users (6 female, age 40-82). The demographics of our participants are shown in Table 2.

Procedure

Following a brief introduction of the study and demographic questions, participants were asked to attach a one dollar bill next to the interface with the goal of facilitating photo taking in the next step to include both the dollar bill and the complete interface into the field of view. Then, participants were asked to take five photos of the microwave control panel with the assistance of the Facade iOS app, followed by another five photos taken with the built-in camera app on iOS. After each photo was taken, simulated crowd feedback on image quality was provided. These images were used to evaluate the perspective transformation and crowdsourced labeling.

Next, the labels for the microwave were entered into the app to simulate it having been crowd labelled, and participants were asked to explore the customization interface for identifying their reading medium and other preferences. Then, based on the reading medium chosen by the participants, the fabricated overlay of the microwave was presented to the participants, and they were asked to attach the overlay onto the microwave with double-sided tapes by aligning the edges. Images of the attached overlay was taken, and experimenters tested individual buttons to evaluate whether the alignment was sufficient for activating the microwave controls.

Next, we used a sheet of Braille or embossed letters in randomized order to familiarize participants with the shape and feeling of the tactile labels. Then, participants were asked to identify and read out the label of each button of the microwave. Accuracy was recorded, and participants were told the meaning of each abbreviation, *e.g.*, BP for Baked Potato.

Next, participants were asked to complete 11 locating tasks and 4 simulated cooking tasks. For locating tasks, the participant was asked to locate a button with the assistance of the tactile overlay, and then push to trigger the button. Tasks included Power Level, Baked Potato, Frozen Dinner, Kitchen Timer, Clock, Popcorn, Time defrost, 0, 2, 4, and 8. For simulated cooking tasks, we designed more realistic tasks that involved a series of button presses. For example, a multi-button cooking task would require pressing a configure button (*e.g.*, weight defrost, time defrost, or time cook), followed by setting a time duration by pressing the number pads (*e.g.*, 2, 1, 0 for two minutes and 10 seconds, or two pounds and 10 oz), and finally pressing the 'Start' button. For both locating and simulated cooking tasks, we measured accuracy and time.

After performing tasks on the microwave with the tactile overlay using their reading medium preferences, we also tested overlays we printed out with other settings, such as the same microwave augmented with Braille or embossed letters, and a fridge interface augmented with embossed letters and symbols.

After each step of the study, we collected Likert scale ratings and subjective feedback from the participants. Finally, we ended the study with a semi-structured interview asking for the participant's comments and suggestions on the Facade system. The study took about one and a half hours and the participants were compensated for $50. The whole study was video and audio recorded for further analysis.

ID	Gender	Age	Occupation	Vision Level	Reading Medium	Smartphone Use
P1	Male	63	Retired	Blind, since birth	Braille, 60 years	iPhone, 7 years
P2	Female	68	Retired	Blind, since birth	Braille, 62 years	iPhone, 8 years
P3	Female	75	Retired	Light perception	Braille, 20 years	No
P4	Male	82	Retired	Blind, since 8 years old	Braille, 75 years	No
P5	Female	46	Unemployed	Blind, since birth	Braille, 42 years	iPhone, 2 months
P6	Male	40	IT professional	Light perception, tunnel vision	Mostly audio	iPhone, 10 months
P7	Female	42	AT consultant	Blind, since birth	Braille, 22 years	iPhone, 2 years
P8	Male	43	Rehab counselor	Blind, since birth	Braille, 36 years	Mostly iPhone, 10 years
P9	Female	58	Retired	Blind, since 1 year old	Braille, 50 years	iPhone, 5 years
P10	Female	61	Retired	Light perception, since birth	Braille, 35 years	iPhone, 6 years
P11	Female	68	Retired	Blind, since birth	Braille, 62 years	iPhone, 1.5 years

Table 2. Participant demographics for our user evaluation with 11 blind users.

Results

We now detail our user study results and summarize user feedback and preferences. For all Likert scale questions, participants were asked to rate along a scale of 1 to 5, where 1 is very negative and 5 is very positive, *e.g.,* 1 for very hard to perform, and 5 for very easy to perform.

Participants spent an average of 30.3 seconds ($SD = 19.1$) to attach the dollar bill and found it very easy to perform ($M = 4.8, SD = 0.41$). For taking photos assisted with the Facade mobile app, participants took an average of 33.6 seconds ($SD = 24.5$) to take each photo, and rated neutral for the difficulty of taking photos ($M = 3.2, SD = 1.3$). The reason why it was not easy was mainly because it required users to hold the device very stable, and there was no direct feedback of where the interface was. However, participants mentioned that after taking a few photos, feedback became easier to follow.

For applying the tactile overlay onto the microwave control panel, it took participants an average of 117.1 seconds ($SD = 83.0$) to attach the overlay, including 2 of the 11 participants failed to attach the overlay correctly (Figure 6). Participants rated it relatively easy to attach the overlay ($M = 3.8, SD = 1.9$). Participants applied the strategy of aligning from the top and using gravity to keep the overlay flat and align towards the bottom. P7 suggested that making the edge of the tactile overlay more distinctive can make it easier to align with the interface. Depending on the size of the buttons, slight offset will not affect using the appliance (such as Figure 6 P5). Furthermore, if the buttons are physically raised, they will also help with aligning the tactile overlay.

Identification Tasks

Ten out of 11 participants chose Braille as their primary reading medium and used our tactile overlay augmented with Braille labels, while P6 used the embossed letter version of tactile overlay. In order to compare participants' performance and report on our Braille quality, we only report the performance of the ten participants who used Braille. For identification tasks, it took participants an average of 112.6 seconds ($SD = 44.1$) to read through all 25 buttons of the microwave, with an accuracy of 98.3% ($SD = 0.018$). Errors happened to letters including C, D, and P. Participants rated reading the Braille as easy ($M = 4.2, SD = 0.92$). The errors were mostly caused by the limited resolution of the printer and some remaining artifacts on the print. We believe this will be further resolved with improvements in 3D printers and printing materials, as detailed in the survey of different materials and printing techniques in the next section. Participants also mentioned that when sitting in front of the microwave in our study, their hands needed to be flipped backward when reading the Braille, which affected the accuracy. This effect will be reduced when they place the microwave at the position they prefer, and as they get familiar with the functions over time as they use them.

Locating and Simulated Cooking Tasks

For locating tasks, it took participants an average of 6.7 seconds ($SD = 4.6$) to locate and push to activate each of the function buttons, while it all took less than 1 second for the number buttons. The overall accuracy was 97.3% ($SD = 0.044$). We asked participants to locate the number pad in a separate task to evaluate whether the different shapes of function and number buttons facilitate locating, and all participants rated it as very easy (5). Participants also found it very easy to remember the button name by acronyms ($M = 4.9, SD = 0.32$), locate the buttons ($M = 4.6, SD = 0.70$), push the buttons ($M = 4.5, SD = 0.53$), as well as operate the microwave with the tactile overlay ($M = 4.8, SD = 0.42$).

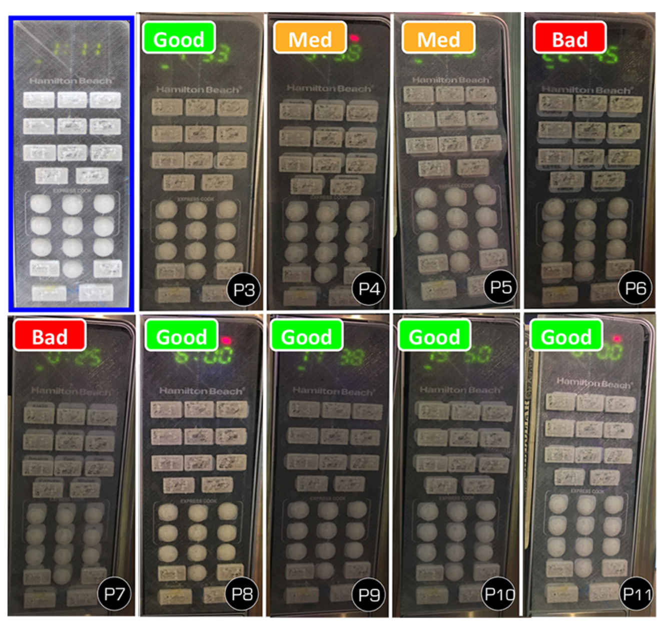

Figure 6. Examples of the attached overlay performed by the participants. For P4 and P5, slight offset did not affect using the appliance. An exact alignment is shown in the top left corner.

Specifically for pushing the buttons, we observed there were 6 times across all 110 locating tasks participants needed to push the button more than once to activate it. Participants commented that with the overlay, they needed to apply slightly more force to activate the button than the original microwave, but it was still very easy to perform. P3 suggested making the buttons thinner and closer to the interface to reduce the force required, similar to a Dymo label tape.

For simulated cooking tasks, it took participants an average of 17.2 seconds ($SD = 10.1$) to complete each sequence, with an accuracy of 92.5% ($SD = 0.169$).

Embossed Letters and Symbols

Though none of the participants use embossed letter or symbol as their primary reading medium, they have mostly encountered them in everyday lives, such as in elevators, doorways, hotel rooms, or restrooms.

For identification tasks on the embossed letter version of tactile overlay for the microwave control panel, it took participants an average of 218.1 seconds ($SD = 132.9$) to complete all 25 buttons. And for a fridge overlay that contains both embossed letters and symbols, it took them an average of 142.0 seconds ($SD = 76.0$) to read through 10 buttons.

Subjective Feedback

When asked which of the three reading medium they preferred (*i.e.*, Braille, embossed letter, symbol), all participants chose Braille, mostly because it aligns with their primary reading medium. P5 mentioned that if living with sighted people or people with partial vision, he could also accommodate with embossed letters.

When asked to compare Facade with the traditional method of applying Braille labels, participants commented *"I like [Facade] much better. I can do it myself, to me it's huge. I don't need to wait for someone to come over and label things for me. If template gets damaged, then I can create a new one. With the [traditional] labels I made, things start get peeled off soon. I think this is neat (P1)"*, and *"This makes a lot more sense. Dymo easily fall off. I like this better (P6)."*

Participants also provided suggestions to make Facade work with interface widgets of other shapes, such as circular knobs (P9). P8 suggested to add a simple/advanced mode in the customization interface in the mobile app for people who prefer labeling the complete panel versus only a small set of buttons. P1 suggested that the feedback provided in the app should be more specific, such as saying "dollar bill in focus", "images are too close", *etc.*

Overall, participants were excited about the potential of Facade and several asked when they can use it on their appliances.

TECHNICAL EVALUATION

We conducted a multi-part technical evaluation in order to understand how each component of the Facade pipeline performs across a range of interfaces and usage scenarios.

Interface Capture

We first evaluated how well the Facade iOS app performs in assisting blind people in capturing photos containing both the dollar bill and the interface of the device, and how well our perspective transformation component performs in warping images to a front view of the interface.

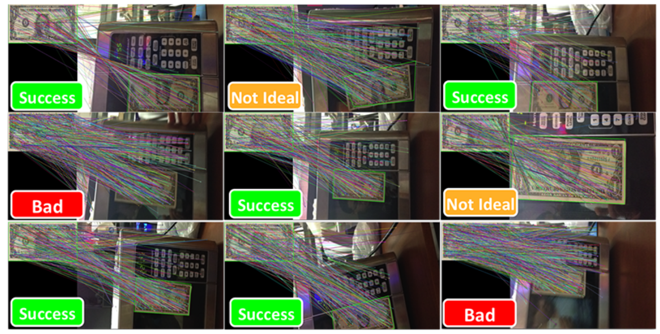

Figure 7. Examples for image localization and warping on photos taken by the participants using a dollar bill as the fiducial marker. Boxes show in green represents warping results that were good enough for generating a usable tactile overlay model, while those shown in orange and red represents failure cases.

We used photos of the microwave taken by the participants from our user evaluations (described in the previous section). Out of the 55 images taken when the Facade iOS app provided feedback, the perspective transformation component was able to identify the dollar bill and successfully warp the image to a front perspective for 34 cases (61.8%). In 4 of 55 cases, it identified the dollar bill, but the resulting warping was not suitable for further labeling and printing. In the remaining 17, the dollar bill failed to be localized. Pictures taken with the Camera app built into iOS were worse. Only 18 (32.7%) were successful, 3 were not ideal, and 34 did not localize the bill.

These image are then labeled by crowd workers, and labels are used to generate 3D models of the tactile overlay. Each segmenting task paid \$0.15 (~40sec of work, \$13.5/hr). Each labeling task paid \$.02 (<10sec, \$9/hr). We evaluated the crowdsourcing workflow, and generated analogous results to [22] as they share a similar crowdsourcing workflow. In prior work, the crowdsourcing labeling workflow was fast (8 minutes), accurate (99.7%), and inexpensive (\$1.15). Accuracy is high because tasks are simple, and we perform automatic and redundancy checks on button size, aspect ratios, and labels. For the 34 successful pictures taken using the Facade app, Facade was able to generate a usable tactile overlay model for 21 of them (61.7%). On the other hand, 16 out of 18 images taken with the built-in Camera app were able to generate a usable overlay model. Figure 7 shows examples for image localization and warping on photos taken by the participants.

The results show our Facade iOS app allowed participants to take better photos for generating the tactile overlay. It is important to note that since the Facade iOS app streams all images to the backend server when aiming the camera, we could configure our system to automatically pick several images where the dollar bill can be found before the user clicks the "take photo" button. This would also require adding another step in the crowdsourcing workflow for the crowd workers to select the best warped image.

Model Production

We next tested our pipeline on several other appliance interfaces, including three different microwaves, two refrigerators,

Material	Printer	Resolution	Price
NinjaFlex	Lulzbot TAZ5	Medium	$10.00
SemiFlex	Lulzbot TAZ5	Medium	$10.00
PolyFlex	Lulzbot TAZ5	Medium	$10.00
Nylon	ProJet 660 SLS	High	$31.40
Flexible Resin	FormLab SLS	High	$21.70

Table 3. We experimented with a variety of materials and printing techniques. Test prints were ordered from 3D Hubs.

and a printer, some of these are shown in Figure 1. In addition to varying appliances, we also experimented with a variety of printing techniques. Figure 5(a-d) shows tactile overlays printed with PrintrBot Simple Maker's kit (FDM) in our lab, each costs less than $5 (printed in 15% infill, 0.4mm layer thickness, 3 solid layers for top/bottom). Figure 5fg shows two example prints ordered from 3D Hubs using PolyFlex and SemiFlex materials. As shown in Table 3, we also tried out Nylon and Flexible Resin printed with SLS (selective laser sintering) printing technique, which generated much higher resolution Braille labels.

DISCUSSION AND FUTURE WORK

Facade enables blind people to access flat physical interfaces by auto-generating a tactile overlay. We focused on augmenting inaccessible buttons in this paper, while this concept can be extended to work with other types of interface elements. For example, for a mechanical knob, tactile markings can be attached to the main interface, leaving the knob area empty, and a separate pointer printed and attached. Another approach is to fabricate a knob that supports internal movement.

Once the original physical interface is covered by the tactile buttons, sighted users living with blind users, or external caretakers cannot easily identify the original functions [8]. We chose transparent filament to print the overlaid buttons to see through the background. Yet, the button had multiple layers, which reduced transparency. To address this, we can support both sighted and blind people to access the appliances by applying different colors of materials to make the text labels visually salient, similar to Figure 5b. For appliances where visual elements are not as cluttered as the examples we show, we could place the tactile labels around the interactive elements and leave a hole for the button to directly make them accessible to sighted people (similar to Thingiverse thing: 1415446). Other attachment strategies can also be investigated with mechanical structures, such as hinge, flip door, sliding, *etc.*

For buttons that are not flat, our current approach of using a flat sheet for attachment wouldn't work. As an initial investigation, we show in the copier example in Figure 1 that additional measurements are required for creating concave structures to fit the embossed buttons. We have implemented this feature as an input parameter in our fabrication design tool. However, more advanced approaches need to be integrated to support the acquisition of this value. For example, instead of asking the blind person to take a photo, using a depth camera could better capture the convex properties of the physical interface.

For interfaces or buttons that are too small or cluttered, putting tactile labels on top of the buttons wouldn't work due to the fixed size of Braille. To mitigate this problem, we could configure Facade to print an overlay with minimal markings to attach to the interface, while generating another legend detailing the interface layout and labels on the side.

Our 3D printed augmentation is designed to overlay an interface which is triggered by manual force. If the augmented sheet covers a capacitive touchscreen, it would likely disrupt operation of the interface. One possible solution to address this problem is to print the button with conductive material that connects human skin's conductivity through the 3D printed objects. While this is an interesting approach and needs to be investigated to expand the range of interface that Facade can support, we leave this for future work.

The cost of Facade is rapidly changing and it may soon be competitive with creating labels with tape. 3D printing material is quickly getting cheaper, and approaching that of embossed labels using Dymo tape. In our current pipeline, the interface layout and labels are generated from the crowdsourcing workflow. However, these could also be acquired from remote friends or family, provided by the appliance manufacturers, or automatically retrieved from online manuals. Collectively, labels for common appliances may also benefit a new user. Furthermore, similar approaches can be used for other tasks, such as for a sighted partner or a building manager to quickly collect images and automatically produce tactile labels and augmentations to make a space accessible, which is likely more efficient than manual labeling.

Our evaluation demonstrates the viability of Facade by deeply evaluating each component. This will guide us (and others) in future work to understand how each component is likely to work in deployment and how we might usefully improve the system (*e.g.*, using more advanced blind photography, using more robust crowd labeling workflows, applying other fabrication techniques or materials, *etc.*)

CONCLUSIONS

We have presented *Facade*, a crowdsourced fabrication pipeline for blind users to augment inaccessible physical interfaces by 3D printed tactile overlays. Our system empowers blind users to access physical interfaces in everyday lives in an independent and inexpensive way. We introduced the design of the system and its technical architecture, evaluated it in a user study with 11 blind participants, and evaluated each component separately to understand its limitations.

Compared with traditional embossed labelers, Facade does not require in-person sighted assistance, provides richer tactile feedback using different reading mediums and button shapes, and reduces memory load by providing a legend and in-app support: embossed labels do none of these. Our research envisions a future when 3D printers are faster and more ubiquitous in people's homes. Facade can benefit blind users by generating tactile overlays to home appliances in minutes, complementing or replacing in-home embossed labelers.

ACKNOWLEDGMENTS

This work has been supported by the National Science Foundation, Google, and the National Institute on Disability, Independent Living, and Rehabilitation Research (NIDILRR). We thank the blind participants and workers on Mechanical Turk who contributed to our studies.

REFERENCES

1. American Foundation for Blind 2016. Braille: Deciphering the code. http://braillebug.afb.org/braille_deciphering.asp. (2016).

2. Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. 2006. SURF: Speeded up robust features. In *Proceedings of the 9th European Conference on Computer Vision - Volume Part I (ECCV'06)*. Springer-Verlag, Berlin, Heidelberg, 404–417. DOI:http://dx.doi.org/10.1007/11744023_32

3. Jeffrey P. Bigham, Michael S. Bernstein, and Eytan Adar. 2015. Human-computer interaction and collective intelligence. *Handbook of Collective Intelligence* 57 (2015).

4. Jeffrey P. Bigham, Chandrika Jayant, Hanjie Ji, Greg Little, Andrew Miller, Robert C. Miller, Robin Miller, Aubrey Tatarowicz, Brandyn White, Samual White, and Tom Yeh. 2010a. VizWiz: Nearly real-time answers to visual questions. In *Proceedings of the 23rd Annual ACM Symposium on User Interface Software and Technology (UIST '10)*. ACM, New York, NY, USA, 333–342. DOI: http://dx.doi.org/10.1145/1866029.1866080

5. Jeffrey P. Bigham, Chandrika Jayant, Andrew Miller, Brandyn White, and Tom Yeh. 2010b. VizWiz::LocateIt - Enabling blind people to locate objects in their environment. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2010 IEEE Computer Society Conference on*. IEEE, 65–72. DOI: http://dx.doi.org/10.1109/CVPRW.2010.5543821

6. Erin Brady and Jeffrey P. Bigham. 2015. Crowdsourcing accessibility: Human-powered access technologies. *Foundations and Trends in Human-Computer Interaction* 8, 4 (2015), 273–372. DOI: http://dx.doi.org/10.1561/1100000050

7. Erin Brady, Meredith Ringel Morris, Yu Zhong, Samuel White, and Jeffrey P. Bigham. 2013. Visual challenges in the everyday lives of blind people. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13)*. ACM, New York, NY, USA, 2117–2126. DOI: http://dx.doi.org/10.1145/2470654.2481291

8. Stacy M. Branham and Shaun K. Kane. 2015. Collaborative accessibility: How blind and sighted companions co-create accessible home spaces. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI '15)*. ACM, New York, NY, USA, 2373–2382. DOI: http://dx.doi.org/10.1145/2702123.2702511

9. 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)*. ACM, New York, NY, USA, 131–138. DOI: http://dx.doi.org/10.1145/2148131.2148160

10. 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)*. ACM, New York, NY, USA, 445–457. DOI:http://dx.doi.org/10.1145/2858036.2858375

11. 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)*. ACM, New York, NY, USA, 525–534. DOI:http://dx.doi.org/10.1145/2702123.2702525

12. Erin Buehler, Amy Hurst, and Megan Hofmann. 2014a. Coming to grips: 3D printing for accessibility. In *Proceedings of the 16th International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS '14)*. ACM, New York, NY, USA, 291–292. DOI: http://dx.doi.org/10.1145/2661334.2661345

13. Erin Buehler, Shaun K. Kane, and Amy Hurst. 2014b. ABC and 3D: Opportunities and obstacles to 3D printing in special education environments. In *Proceedings of the 16th International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS '14)*. ACM, New York, NY, USA, 107–114. DOI: http://dx.doi.org/10.1145/2661334.2661365

14. H. Burton, D.G. McLaren, and R.J. Sinclair. 2006. Reading embossed capital letters: An fMRI study in blind and sighted individuals. *Human brain mapping* 27, 4 (2006), 325–339. DOI: http://dx.doi.org/10.1002/hbm.20188

15. Xiang 'Anthony' Chen, Stelian Coros, Jennifer Mankoff, and Scott E. Hudson. 2015. Encore: 3D printed augmentation of everyday objects with printed-over, affixed and interlocked attachments. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software and Technology (UIST '15)*. ACM, New York, NY, USA, 73–82. DOI:http://dx.doi.org/10.1145/2807442.2807498

16. Xiang 'Anthony' Chen, Jeeeun Kim, Jennifer Mankoff, Tovi Grossman, Stelian Coros, and Scott E. Hudson. 2016. Reprise: A design tool for specifying, generating, and customizing 3D printable adaptations on everyday objects. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology (UIST '16)*. ACM, New York, NY, USA, 29–39. DOI: http://dx.doi.org/10.1145/2984511.2984512

17. Adrian A. de Freitas, Michael Nebeling, Xiang 'Anthony' Chen, Junrui Yang, Akshaye Shreenithi Kirupa Karthikeyan Ranithangam, and Anind K. Dey. 2016. Snap-To-It: A user-inspired platform for opportunistic device interactions. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. ACM, New York, NY, USA, 5909–5920. DOI: http://dx.doi.org/10.1145/2858036.2858177

18. Martin A. Fischler and Robert C. Bolles. 1981. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM* 24, 6 (June 1981), 381–395. DOI: http://dx.doi.org/10.1145/358669.358692

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)*. ACM, New York, NY, USA, 81–90. DOI: http://dx.doi.org/10.1145/2982142.2982163

20. Timo Götzelmann and Aleksander Pavkovic. 2014. *Towards automatically generated tactile detail maps by 3D printers for blind persons*. Springer International Publishing, Cham, 1–7. DOI: http://dx.doi.org/10.1007/978-3-319-08599-9_1

21. Anhong Guo, Xiang 'Anthony' Chen, and Jeffrey P. Bigham. 2015. ApplianceReader: A wearable, crowdsourced, vision-based system to make appliances accessible. In *Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems (CHI EA '15)*. ACM, New York, NY, USA, 2043–2048. DOI: http://dx.doi.org/10.1145/2702613.2732755

22. Anhong Guo, Xiang 'Anthony' Chen, Haoran Qi, Samuel White, Suman Ghosh, Chieko Asakawa, and Jeffrey P. Bigham. 2016a. 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)*. ACM, New York, NY, USA, 651–664. DOI: http://dx.doi.org/10.1145/2984511.2984518

23. Anhong Guo, Jeeeun Kim, Xiang 'Anthony' Chen, Tom Yeh, Scott E. Hudson, Jennifer Mankoff, and Jeffrey P. Bigham. 2016b. Facade: Auto-generating tactile interfaces to appliances. In *Proceedings of the 18th International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS '16)*. ACM, New York, NY, USA, 315–316. DOI: http://dx.doi.org/10.1145/2982142.2982187

24. 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)*. ACM, New York, NY, USA, 1769–1780. DOI: http://dx.doi.org/10.1145/2858036.2858340

25. Chandrika Jayant, Hanjie Ji, Samuel White, and Jeffrey P. Bigham. 2011. Supporting blind photography. In *The Proceedings of the 13th International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS '11)*. ACM, New York, NY, USA, 203–210. DOI: http://dx.doi.org/10.1145/2049536.2049573

26. 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)*. ACM, New York, NY, USA, 247–252. DOI: http://dx.doi.org/10.1145/2538862.2538975

27. 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. DOI: http://dx.doi.org/10.1145/2702123.2702144

28. Benjamin Lafreniere, Tovi Grossman, Fraser Anderson, Justin Matejka, Heather Kerrick, Danil Nagy, Lauren Vasey, Evan Atherton, Nicholas Beirne, Marcelo H. Coelho, Nicholas Cote, Steven Li, Andy Nogueira, Long Nguyen, Tobias Schwinn, James Stoddart, David Thomasson, Ray Wang, Thomas White, David Benjamin, Maurice Conti, Achim Menges, and George Fitzmaurice. 2016. Crowdsourced fabrication. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology (UIST '16)*. ACM, New York, NY, USA, 15–28. DOI:http://dx.doi.org/10.1145/2984511.2984553

29. Walter S. Lasecki, Phyo Thiha, Yu Zhong, Erin Brady, and Jeffrey P. Bigham. 2013. Answering visual questions with conversational crowd assistants. In *Proceedings of the 15th International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS '13)*. ACM, New York, NY, USA, Article 18, 8 pages. DOI: http://dx.doi.org/10.1145/2513383.2517033

30. Roberto Manduchi and James M. Coughlan. 2014. The last meter: Blind visual guidance to a target. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '14)*. ACM, New York, NY, USA, 3113–3122. DOI: http://dx.doi.org/10.1145/2556288.2557328

31. Samantha McDonald, Niara Comrie, Erin Buehler, Nicholas Carter, Braxton Dubin, Karen Gordes, Sandy McCombe-Waller, and Amy Hurst. 2016. Uncovering challenges and opportunities for 3D printing assistive technology with physical therapists. In *Proceedings of the 18th International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS '16)*. ACM, New York, NY, USA, 131–139. DOI: http://dx.doi.org/10.1145/2982142.2982162

32. 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 and Accessibility (ASSETS '14)*. ACM, New York, NY, USA, 275–276. DOI:http://dx.doi.org/10.1145/2661334.2661392

33. George A. Miller. 1956. The magical number seven, plus or minus two: Some limits on our capacity for processing information. *Psychological review* 63, 2 (1956), 81. DOI: http://dx.doi.org/10.1037/h0043158

34. Marius Muja and David G. Lowe. 2009. Fast approximate nearest neighbors with automatic algorithm configuration. In *International Conference on Computer Vision Theory and Application VISSAPP'09)*. INSTICC Press, 331–340.

35. Jeffrey Nichols, Brad A. Myers, Michael Higgins, Joseph Hughes, Thomas K. Harris, Roni Rosenfeld, and Mathilde Pignol. 2002. Generating remote control interfaces for complex appliances. In *Proceedings of the 15th Annual ACM Symposium on User Interface Software and Technology (UIST '02)*. ACM, New York, NY, USA, 161–170. DOI:http://dx.doi.org/10.1145/571985.572008

36. Raf Ramakers, Fraser Anderson, Tovi Grossman, and George Fitzmaurice. 2016. RetroFab: A design tool for retrofitting physical interfaces using actuators, sensors and 3D printing. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. ACM, New York, NY, USA, 409–419. DOI: http://dx.doi.org/10.1145/2858036.2858485

37. 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)*. ACM, New York, NY, USA, 4896–4907. DOI: http://dx.doi.org/10.1145/2858036.2858507

38. Saiganesh Swaminathan, Thijs Roumen, Robert Kovacs, David Stangl, Stefanie Mueller, and Patrick Baudisch. 2016. Linespace: A sensemaking platform for the blind. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. ACM, New York, NY, USA, 2175–2185. DOI: http://dx.doi.org/10.1145/2858036.2858245

39. 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)*. ACM, New York, NY, USA, 71–79. DOI: http://dx.doi.org/10.1145/2982142.2982167

40. United States Access Board 2016. Advancing full access and inclusion for all. https://www.access-board.gov/guidelines-and-standards. (2016).

41. Gregg Vanderheiden and Jutta Treviranus. 2011. Creating a global public inclusive infrastructure. In *International Conference on Universal Access in Human-Computer Interaction*. Springer, 517–526. DOI: http://dx.doi.org/10.1007/978-3-642-21672-5_57

42. Marynel Vázquez and Aaron Steinfeld. 2014. An assisted photography framework to help visually impaired users properly aim a camera. *ACM Transactions on Computer-Human Interaction* 21, 5, Article 25 (Nov. 2014), 29 pages. DOI:http://dx.doi.org/10.1145/2651380

43. Samuel White, Hanjie Ji, and Jeffrey P. Bigham. 2010. EasySnap: Real-time audio feedback for blind photography. In *Adjunct Proceedings of the 23rd Annual ACM Symposium on User Interface Software and Technology (UIST '10)*. ACM, New York, NY, USA, 409–410. DOI: http://dx.doi.org/10.1145/1866218.1866244

44. Wikipedia. 2016. Homography (computer vision) — Wikipedia, The Free Encyclopedia. (2016). https://en.wikipedia.org/w/index.php?title=Homography_(computer_vision)&oldid=752434320 [Online; accessed 1-December-2016].

45. Yu Zhong, Pierre J. Garrigues, and Jeffrey P. Bigham. 2013. Real time object scanning using a mobile phone and cloud-based visual search engine. In *Proceedings of the 15th International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS '13)*. ACM, New York, NY, USA, Article 20, 8 pages. DOI: http://dx.doi.org/10.1145/2513383.2513443

46. Yu Zhong, Walter S. Lasecki, Erin Brady, and Jeffrey P. Bigham. 2015. RegionSpeak: Quick comprehensive spatial descriptions of complex images for blind users. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI '15)*. ACM, New York, NY, USA, 2353–2362. DOI: http://dx.doi.org/10.1145/2702123.2702437