Accessible Art Team

Lucy Jiang, Computer Science and Engineering
Tina Li, Mechanical Engineering

Mentored by George Zatloka
June 7, 2019
Executive Summary

Art provides people with a form of communication that fosters creativity, curiosity, and critical thinking — it evokes emotions, inspires others, and empowers creators in a unique way. Our challenge is to develop an accessible, collaborative, affordable, and participatory project that expands access to opportunities for creating and experiencing art to people of all abilities. Over the course of six months, our team designed a micro-installation which features tactile and auditory elements, making art more accessible to people who are blind or visually impaired. Although it took time to narrow down our idea, we focused on methods of making art as interactive as possible. We prototyped circuits, apps, and other mechanisms extensively, and we tested multiple times with people in our primary user group to garner important feedback. We were able to iterate upon our design using user feedback as well, adapting the project to be even more accessible to people who are blind or visually impaired. In June, we hosted a micro-installation event at a local venue to demonstrate our project idea and collect more feedback from students at the University of Washington. Ultimately, our hope is to provide a new platform for accessible art to the community.
The Challenge

Our challenge is to create a platform for people of all abilities to create and experience art and to make art accessible for everybody. The challenge was not proposed by a specific individual; rather, it addresses the universal necessity for people of all abilities to have chances to create and experience art in a new way.

Art is an integral part of the human experience, yet it remains inaccessible to many. One main benefit of creating art is that it gives the creator a sense of independence and empowerment, and one main benefit of experiencing art is that it evokes powerful emotions. Furthermore, art supports critical thinking and can often transcend language and age barriers, appealing to people of all ages, abilities, and nationalities — it is a universal thread that ties humans together.

The primary stakeholders of our project are people who are blind or visually impaired, as they are the primary users of the platform. Our project aims to empower them and instill independence. The secondary stakeholders are museum directors, program coordinators, organizations for accessible art, and funders. They are responsible for supporting, promoting, and funding or adapting our platform; they will provide opportunities for people of all abilities create and experience art.

Our point of view (POV) statement is:

*People of all abilities, especially people who are blind or visually impaired later on in life, NEED to have access, participation, and interaction to create art regularly BECAUSE art sparks curiosity, showcases individuality, and empowers artists and participants."

Currently, platforms that allow people who are blind or visually impaired to experience art include relief art and sound painting. Existing solutions focus on senses of touch, hearing, and taste, and the activities are generally more hands-on. However, many technologies such as 3D printers can make art more tactile, but they are inaccessible for people who are blind or visually impaired to create art.

Additionally, activities such as sculpting provide opportunities for people who are blind or visually impaired to create and experience art, but common challenges include its cost and difficulties in determining an artist’s progress with their art piece. Therefore, our team is designing a way to create art that solves the limitations that current activities have, such as generating feedback while creating art; we hope to fill the unique gap in accessible art that is left by existing solutions.

The Solution

Our solution is a micro-installation which uses tactile and auditory elements to make art more accessible to people who are blind or visually impaired. The micro-installation has three stations: 1) Raspberry Pi Capacitive Touch Sensing (Appendix A), 2) Wire Sculpting (Appendix B), and 3) App and External Outputs (Appendix C). At the first station, users choose a textured pipe cleaner. To help distinguish the wires, the three senses of sight,
touch, and hearing are used (Fig. 1). At the second station, users create art by shaping the pipe cleaner chosen from the previous station and inserting the pipe cleaner into the corrugated cardboard (Fig. 2). At the third station, users open the Composer’s Sketchpad app and place the frames they designed from the previous station on top of the iPad Air 2, then trace the contour of the pipe cleaner (Fig. 3). The app plays music based on the wire’s shape and triggers the fan and the heat lamp based on the frequency of the music (Fig. 4). The multi-sensory elements embedded throughout our micro-installation allow people of all abilities to create and experience art.

![Fig. 1. Tactile, auditory, and visual elements that help users choose a pipe cleaner.](image1)

![Fig. 2. Pipe cleaner art from participants.](image2)

![Fig. 3. Tracing the wire’s shape on the app.](image3)

![Fig. 4. Fan and heat lamp external outputs.](image4)

**The Process**

Arriving at our final solution took trial and error. We brainstormed many ideas, but it was challenging to decide on one idea that met our criteria of: the ability to create and experience art, use of technology, feedback while creating art, a low learning curve activity, a hands-on activity, ease of execution, and excitement from the team. We learned that we were not making any progress when brainstorming and prototyping different ideas since we did not have a clear focus (Appendix D). Thinking systematically about what we want the users to do, thinking about the different inputs and outputs we want in our device, and brainstorming easily malleable materials helped us find our direction. We decided on our final design based on its feasibility and complexity, the controllability of the medium (i.e. the pipe cleaner), and its adherence to our criteria.

Once we decided on our final idea, we encountered many problems with manufacturing, our design, and app development. One of the main challenges we had was manufacturing the braille dots for the first station. We looked into 3D printing, but the process was too long and had a low resolution because the dots were so small. To solve this problem, we looked into laser cutting with wood. We realized that the dots broke off due to the high heat the laser was emitting to raise the dots. We then looked into printing on acrylic, but the material was too expensive. Based on our prototypes, we decided to laser cut the dots and insert sewing ball pins to create the braille element on our first station. Other manufacturing and design issues included deciding where the user should touch to trigger the instruments, deciding whether to use a MOSFET or relay, creating a fixed workspace for the user, and figuring out what external outputs should be included and how to trigger them. Some issues we had while developing the app were creating an app that works on both iOS and Android, figuring out how to add music when drawing on the app, and much more. Conducting research and asking experts for help assisted us in solving these problems.
The Future

For this project, we prototyped, tested, and conducted consumer surveys to refine and clarify our ideas. We worked with hardware and software to create aspects that participants did not see, and we also designed and created the elements that participants directly touched and heard. One of the strengths that we identified about our project through prototyping was that the many ways of distinguishing instruments made it much easier for people of all user groups to select their instrument of choice. Braille labels, colors, wire textures, and laser-cut instruments made the first station accessible to people who are blind or visually impaired, people with cognitive impairments, and children.

We were able to identify additional strengths while hosting the micro-installation (Appendix E) and while presenting at the design showcase (Appendix F). The third station, which used the user’s audio input to trigger mechanisms such as the heat lamp and the fan, was well-received by participants. The addition of sensory elements such as heat and cold air made it more accessible for people who may not be able to hear the entire range of sounds that they are creating, as they were able to feel the different temperatures and distinguish the high and low pitches of their art. However, one major challenge regarding the third station was its connectivity to the app — the system performed significantly better when plugged into a computer rather than the iPad with the app due to the greater frequency variations in songs rather than the participants’ creations. Moving forward, we hope to improve this system and make it more reliable so that it will work well with any device and in any venue. Another improvement is to make the entire micro-installation more compact, as it required a substantial amount of hardware which was difficult to transport.

Significant unknowns to this project include developing a multi-platform app; while we began development on an Android app using React Native, there was not enough time to finish the functionality and it was difficult to export the app onto iOS platforms. Some additional features we considered adding to the app include automatically recognizing the color of a participant’s pipe cleaner, adjusting the volume based on the pressure of a participant’s tracing finger, and using multiple pipe cleaners simultaneously to create a symphony of sounds. While this level of app development is complex, we hope to continue learning techniques that will make it possible for us to work on more of these features.

We tested our design with two of our friends who are blind or visually impaired and garnered extremely useful feedback from both participants. While we collected data in the form of a survey with optional explanation fields, we hope to continue testing with people in our primary user group. This would be a great place to start for the team continuing this project. It is imperative that we gain even more insight about how people who are blind or visually impaired can use our project, and that we iterate on the project as necessary. One important piece of feedback was that using colors to distinguish the pipe cleaners could be confusing for people who are blind, as it adds “another wrinkle to the design.” We have since altered our design to take this feedback into account — while we have kept the same pipe cleaners with five distinct colors, we will alter the braille labels to say what instrument the pipe cleaners correspond to instead of saying the color of the pipe cleaner.

The pressing needs of this project are to conduct more testing, to improve the reliability of the heat lamp and fan, and to develop a more accessible app. Even further on, we hope to adapt the project so that it is more accessible to people with mobility impairments or other diverse abilities so that art can truly be accessible to all.
References
N/A
Appendix A: 1st Station — Raspberry Pi Capacitive Touch Sensing

I. Additional Information

The first station is a pipe cleaner holder, consisting of five tubes wrapped in aluminium foil that hold the different pipe cleaners (Fig. 1). The tubes are positioned using laser cut-outs in the wooden board. The wooden board has five different colors written in braille and the shapes of five different instruments engraved on the surface.

The purpose of the first station is to augment the overall experience of this project. At the first station, participants will choose a pipe cleaner to use in the next station. To help them choose a pipe cleaner, the three senses of sight, touch, and hearing are addressed. Each pipe cleaner has a distinct color, texture, and sound that makes it unique (Table 1). The sound is triggered with the use of the Raspberry Pi B and the Adafruit Capacitive Touch HAT (Appendix A - Schematic Design). When a certain tube is touched, the corresponding instrument will play. A user can choose a pipe cleaner based on its color, instrument, or texture. Because color is inaccessible to people who are blind or visually impaired, the braille on the board identifies which color they choose.

Table 1. Pipe cleaner attributes (color, texture, and instrument).

<table>
<thead>
<tr>
<th>Color</th>
<th>Texture</th>
<th>Instrument</th>
</tr>
</thead>
<tbody>
<tr>
<td>Orange</td>
<td>Jumbo</td>
<td>Violin</td>
</tr>
<tr>
<td>Green</td>
<td>Regular</td>
<td>Guitar</td>
</tr>
<tr>
<td>Blue</td>
<td>Wire</td>
<td>Flute</td>
</tr>
<tr>
<td>Red</td>
<td>Bumpy</td>
<td>Piano</td>
</tr>
<tr>
<td>Yellow</td>
<td>Rough</td>
<td>Saxophone</td>
</tr>
</tbody>
</table>

II. Schematic Diagram
# Code edited by Tina Li; only changes are the sounds played at each port (sounds downloaded from http://www.findsounds.com)
# Copyright (c) 2014 Adafruit Industries
# Author: Tony DiCola
#
# Permission is hereby granted, free of charge, to any person obtaining a copy
# of this software and associated documentation files (the "Software"), to deal
# in the Software without restriction, including without limitation the rights
# to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
# copies of the Software, and to permit persons to whom the Software is
# furnished to do so, subject to the following conditions:
#
# The above copyright notice and this permission notice shall be included in
# all copies or substantial portions of the Software.
#
# THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
# IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
# FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
# AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
# LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
# OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
# THE SOFTWARE.

import sys
import time
import pygame
import Adafruit_MPR121.MPR121 as MPR121

# Thanks to Scott Garner & BeetBox!
# https://github.com/scottgarner/BeetBox/

print('Adafruit MPR121 Capacitive Touch Audio Player Test

# Create MPR121 instance.
cap = MPR121.MPR121()

# Initialize communication with MPR121 using default I2C bus of device, and
# default I2C address (0x5A). On BeagleBone Black will default to I2C bus 0.
if not cap.begin():
    print('Error initializing MPR121. Check your wiring!')
    sys.exit(1)

# Alternatively, specify a custom I2C address such as 0x5B (ADDR tied to 3.3V),
# 0x5C (ADDR tied to SDA), or 0x5D (ADDR tied to SCL).
#cap.begin(address=0x5B)
#
# Also you can specify an optional I2C bus with the bus keyword parameter.
#cap.begin(busnum=1)

pygame.mixer.pre_init(44100, -16, 12, 512)
pygame.init()

# Define mapping of capacitive touch pin presses to sound files
# tons more sounds are available but because they have changed to .flac in /opt/sonic-pi/etc/samples/ some will not work
# more .wav files are found in /usr/share/scratch/Media/Sounds/ that work fine this example uses Animal sounds.

SOUND_MAPPING = {
    0: '/home/pi/Music/violin.wav',
    1: '/home/pi/Music/chord-F.wav',
    2: '/home/pi/Music/Flutehi.wav',
    3: '/home/pi/Music/sound2.wav',
    4: '/home/pi/Music/sax.wav',
}

sounds = [0,0,0,0,0,0,0,0,0,0,0,0]

for key,soundfile in SOUND_MAPPING.iteritems():
    sounds[key] = pygame.mixer.Sound(soundfile)
    sounds[key].set_volume(1);

# Main loop to print a message every time a pin is touched.
print('Press Ctrl-C to quit.')
last_touched = cap.touched()
while True:
    current_touched = cap.touched()
    # Check each pin's last and current state to see if it was pressed or released.
    for i in range(12):
        # Each pin is represented by a bit in the touched value. A value of 1
        # means the pin is being touched, and 0 means it is not being touched.
        pin_bit = 1 << i
        # First check if transitioned from not touched to touched.
        if current_touched & pin_bit and not last_touched & pin_bit:
            print('{0} touched!'.format(i))
            if (sounds[i]):
                sounds[i].play()
        # Then check if transitioned from touched to not touched.
        if not current_touched & pin_bit and last_touched & pin_bit:
            print('{0} released!'.format(i))
    # Update last state and wait a short period before repeating.
    last_touched = current_touched
    time.sleep(0.1)

    # Alternatively, if you only care about checking one or a few pins you can
    # call the is_touched method with a pin number to directly check that pin.
    # This will be a little slower than the above code for checking a lot of pins.
    #if cap.is_touched(0):
    #    print('Pin 0 is being touched!')

    # If you're curious or want to see debug info for each pin, uncomment the
    # following lines:
    #print('0x{0:0X}'.format(cap.touched()))
    #filtered = [cap.filtered_data(i) for i in range(12)]
    #print('Filt:', '	'.join(map(str, filtered)))
    #base = [cap.baseline_data(i) for i in range(12)]
    #print('Base:', '	'.join(map(str, base))