Drawing-triggered Music Generation Application

Yu Qin
yuqin@umich.edu

Design Strategy

This interface links the behavior of drawing with sound generation. Drawing one line or one triangle would trigger different sounds. Drawing line will trigger cello sound playing while triangle cause bell knocking, which maps the feeling of shape with the vibe of the sound; Line feels mild while the triangle feels tinkling. This application will keep capture the audio signal to control the sounds playing. Users can draw multiple lines and triangles to control sound playing in real time.
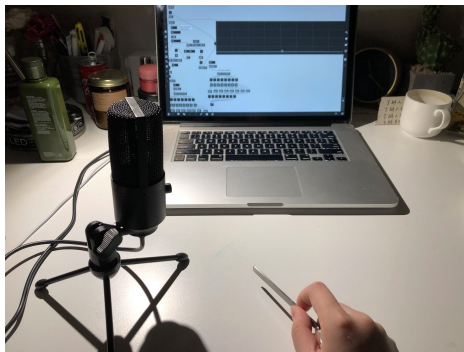
User Interface Design

Figure 1. The hardware set up.
As displayed in Figure one, users are allowed to draw the pattern via a metal bar on the surface of the table to trigger the program.

Program Design

Audio Capturing
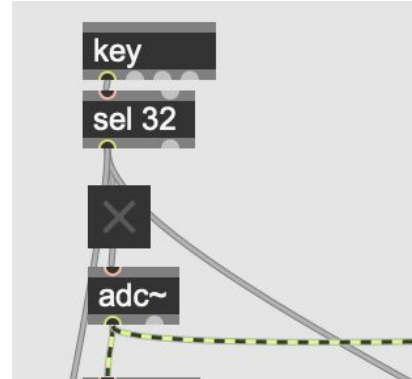The user needs to press the key to start audio capturing.

Figure 2. Max interface to show how to start capturing the audio signal by pressing the space.

Shape Detection
To differentiate the movements of the drawing line and drawing triangles, I visualized the audio signals respectively for two shapes. I found the essential difference between the line drawing signal and the triangle signal is the peak intervals. Drawing triangle causes bigger intervals between peaks than drawing lines. What's more, I noticed that the peak happened when I draw the corner of the shape and there is no corner for a line so the wave of line drawing is more smooth than the wave of triangle drawing and thus smaller intervals between peaks.
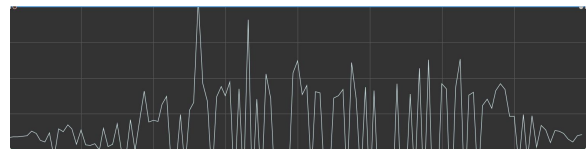

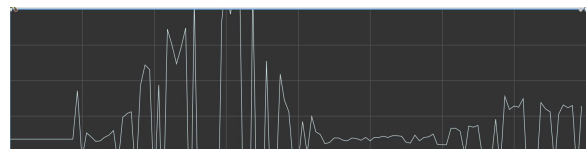
Figure 3. line drawing audio signal.

Figure 4. triangle drawing audio signal. What I did in the signal processing is I used the Max Object fzero to estimate the fundamental frequency of an incoming, monophonic audio signal. It performs multiple layers of wavelet transforms on an incoming vector, comparing the spacing between the peaks in each. I used the value of the peak amplitude of the last period to differentiate when the user is drawing the line and when the user is changing the direction of movement to draw the corner of the triangle. As I said above, drawing triangle will cause larger peak amplitude and a larger interval between peak so I use these two parameters to detect whether the user is drawing line or triangle. When the peak is larger than 0.9, it will trigger the triangle track. When the peak is larger than 0.3 while smaller than 0.9, it will trigger the line track.
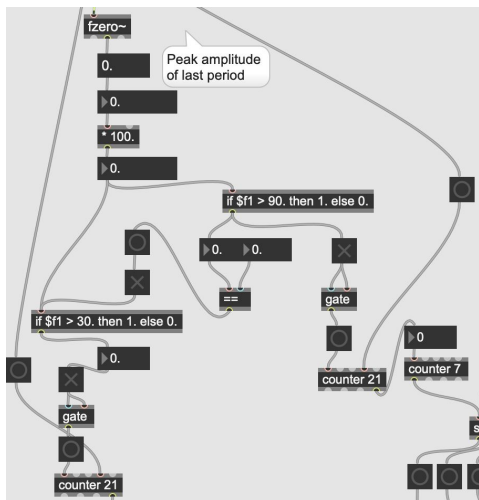

Figure 5. The branch of the program to detect the shape.

Sound Playing
Users can draw multiple lines and triangles to control sound playing in real time. To control the interval of shape detection, this program will first count how many times the audio signal pass the value that differentiates the line and corner, which I found through multiple experiments. Then, I designed a gate signal structure to trigger the sound playing in a drum pattern.
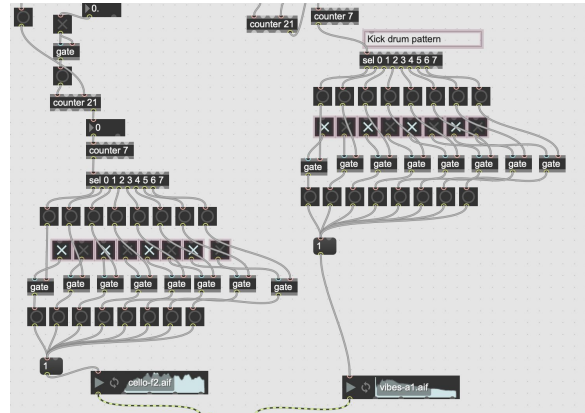

Figure 6. The branch of the program to play sounds.

Result and Discussion

During testing, I found that the location of the microphone will affect the detecting accuracy. Better accuracy happened when putting the Mic right in front of the drawing place. Additionally, the corner detection is not that accurate compared to the line detection because the peak caused by the corner drawing is not consistent. I think if I have more time, I might analyze the shape of the waveform to better classify these two shapes and I think this pattern can be extended to detect more shapes to trigger the sound.

Video

I uploaded the project video to the Vimeo. You can view the video via this link https://vimeo.com/316739549