
ChromaBeat
MAT 236
Taneesha Panda

Inspiration

- Golan Levin - *YellowTail*
- Eric Rosenbaum - *Singing Fingers*
- Sam Bourgault - *Megafauna*

project 1: simple audio-reactive drawing tool

Concept

What creative possibilities arise when sensor data becomes a control parameter for digital drawing?

Does embedding sound and gravity into the control logic of a drawing system create a fundamentally different category of digital art than traditional drawing tools?

- How does it differ from gesture-based systems that track human movement rather than sensor-based tilt orientation?

Motivating Questions

How do I implement specific tilt configurations that produce repeatable visual styles?

How clearly can users perceive the relationship between orientation, frequency band selection, and brush stroke behavior?

Does this system support intentional composition or improvisational exploration?

Sketch

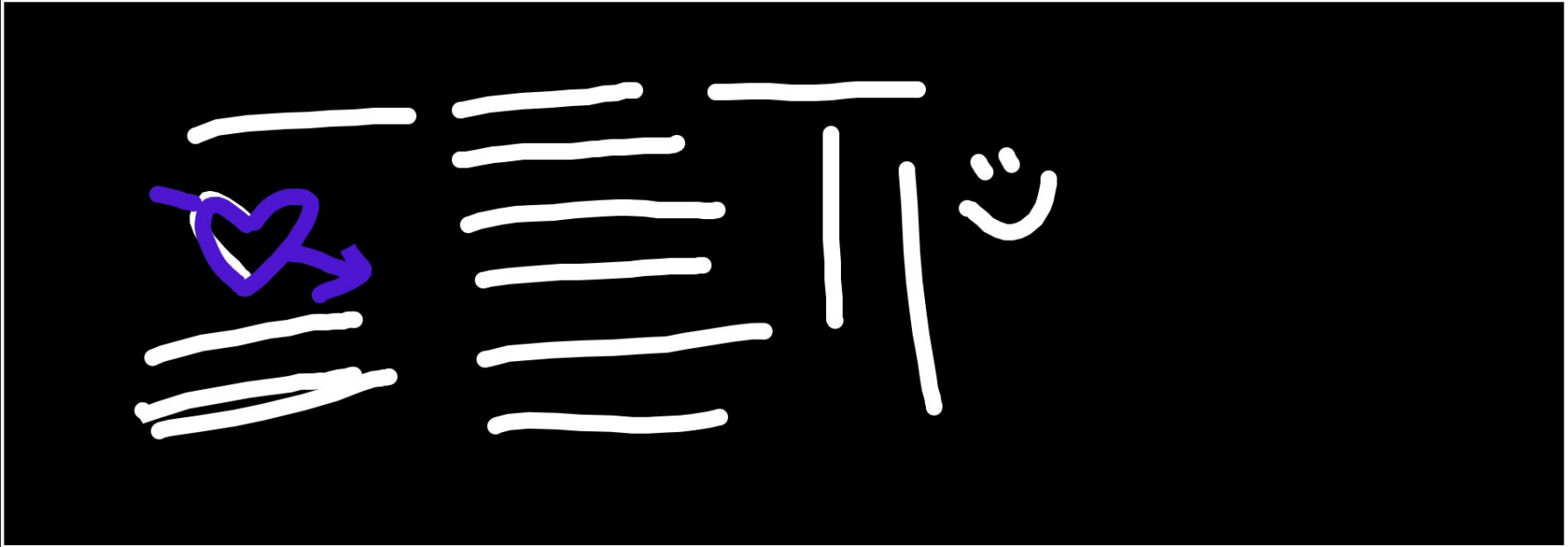
127.0.0.1:5500/index.html

Summarize Chat

Bookmarks Imported New Tab (3) 27) Igneous Ro... Gmail PUSD Home Kundan jewelry ind... Maps YouTube 15 Of The Most Int... 2019-2020 Class A... OneLook Reverse... Carry On (novel) -...

ChromaBeat

Choose File chillmix.m4a



1. Upload audio file (.mp3, .wav, .m4a)
2. Select color
3. Click and drag to draw
4. Tilt ADXL335 along X, Y, or Z plane to change brush size
5. Press C to clear canvas

Type here to search

71°F Sunny ENG 12:10 PM 12/9/2025

ChromaBeat Parameters

User-defined

- select from color picker
- paint across canvas with mousepad, clear with key press

Audio-defined

- brush size based on frequency range
- brush stroke pulsations

Analog Sensor Input

- voltage X,Y,Z values converted to gravity force (-1,0,1)

Technical Considerations

1. Sensor Calibration

- Raw voltage values

- approximate G-force

- classify orientation

2. Drawing System

- Stores strokes as list of points

- Perpendicular vectors with variable width

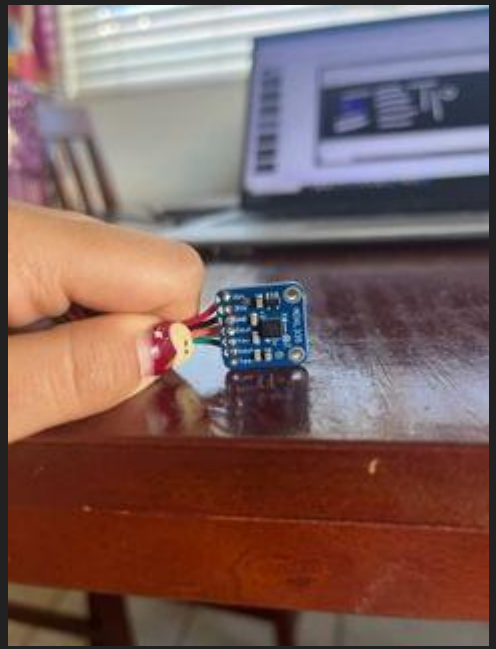
3. WebSocket

- Accelerometer data streamed via WebSocket

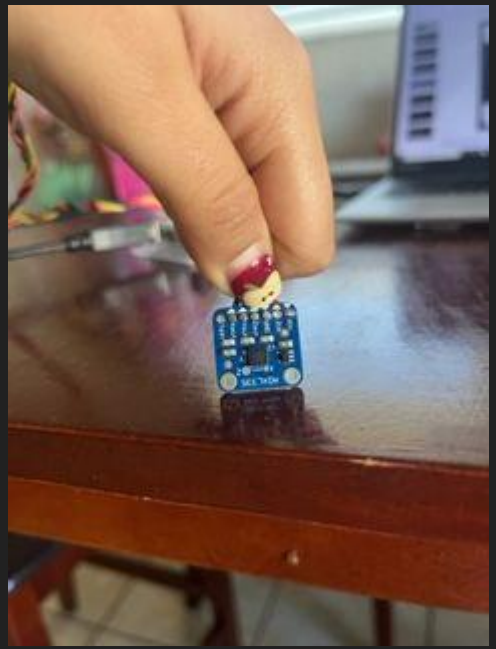
- X/Y/Z positive tilt selects FFT band to change brush size

Defining orientation control with ADXL335

X= +1 → bass



Y= +1 → mid



Z= +1 → treble

Challenges

- Real time sensor input with serial connection
  - Sensor messages arrive asynchronous

    → calculated stable orientation values

    → better parsing logic to prevent rapid flickering
- Poor UI - required a lot of minor adjustments
  - Eg. faulty audio controls

Evolution

Introducing **sensor-based orientation control** transformed the system into a performative, embodied interface.

- Shifted from a visualization tool towards an instrument-like system

My focus evolved from:

- “How can audio input act as drawing material, rather than just background noise?”

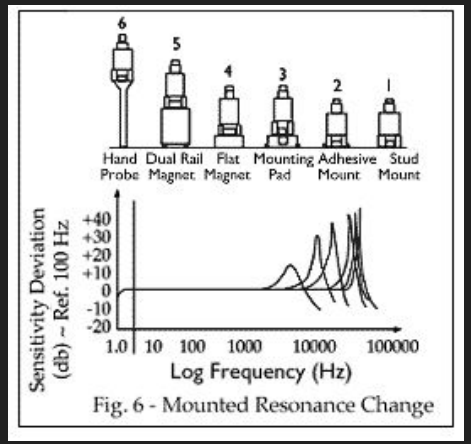
to:

- “How can users manipulate drawing through sound and sensor control?”

Future Directions

Replace mouse-based drawing with the accelerometer itself

- Mount the accelerometer on a flat surface → move it across XY plane
- Directly map coordinates to the p5.js canvas with position tracking from raw accelerometer data



Source: PCB Piezotronics. *Sensor Mounting Techniques.*