Synthesthesia

By DelayLama

Juan Braun
Riccardo Di Bella
Sofia Parrinelli
Ahmed Said
Lea Bian

Introduction

Synesthesia is a neurological phenomenon where one sensory experience is accompanied by another.

One of the most common type of synesthesia is chromesthesia, in which sound automatically evokes an experience of color, shape, and movement

We designed our interpretation of chromesthesia where the color is related to a particular harmonization

How to use Synesthesia

Final device is a closed box with:

- color turntable on the top
  - spin the colour wheel to select the type of chord and to vary the center frequency of a bandpass filter
- the distance sensor sensing on one side
  - control the note by changing the distance of your hand/proper flat surface to the sensor

pictures

Sensors

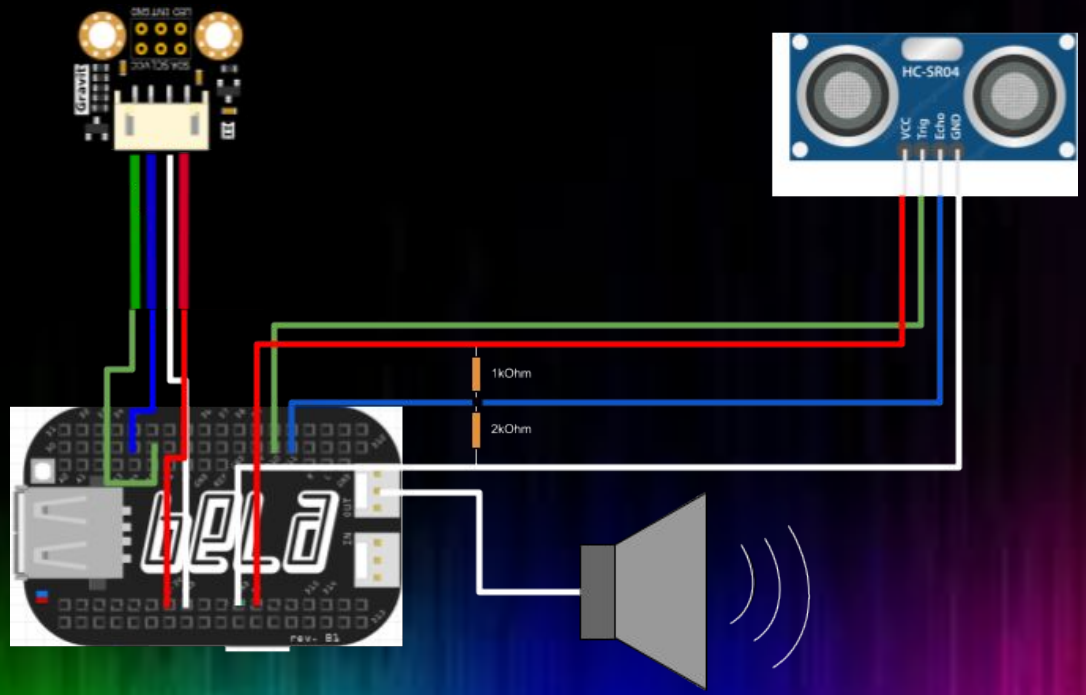
The system uses Bela as the main platform, and two sensors to capture the musician gestures:

RGB Color Sensor
It measures the light in the red, green and blue wavelengths,recognizing the exposed color

Ultrasonic Sensor
Used to control the pitch. It measures the distance of objects placed in front of the sensor

Connection diagram

Bela	Sensor
SCL	RGB - SCL
SDA	RGB - SDA
3.3V	RGB - Vcc
GND	RGB - GND
D-10	HC-SR04 -Trigger
D-11	HC-SR04 - Echo(after voltage divider)
5V	HC-SR04 - Vcc
GND	HC-SR04 - GND



Harmony generation

After receiving the (R,G,B) values a type of chord is selected based on some thresholds

Red: Major chord

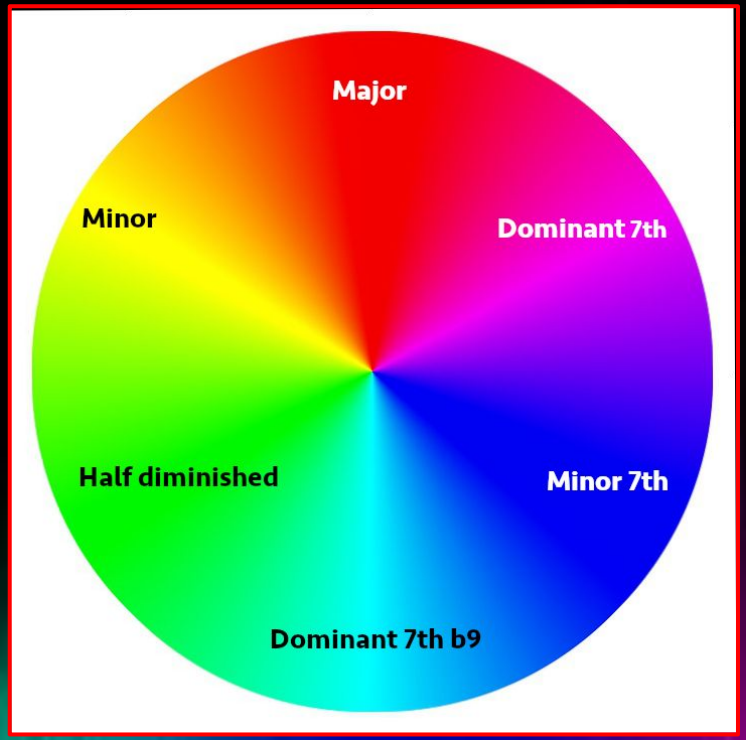
Yellow: Minor chord

Green: Half diminished chord

Teal: Dominant 7th b9 chord

Blue: Minor 7th chord

Purple: Dominant 7th

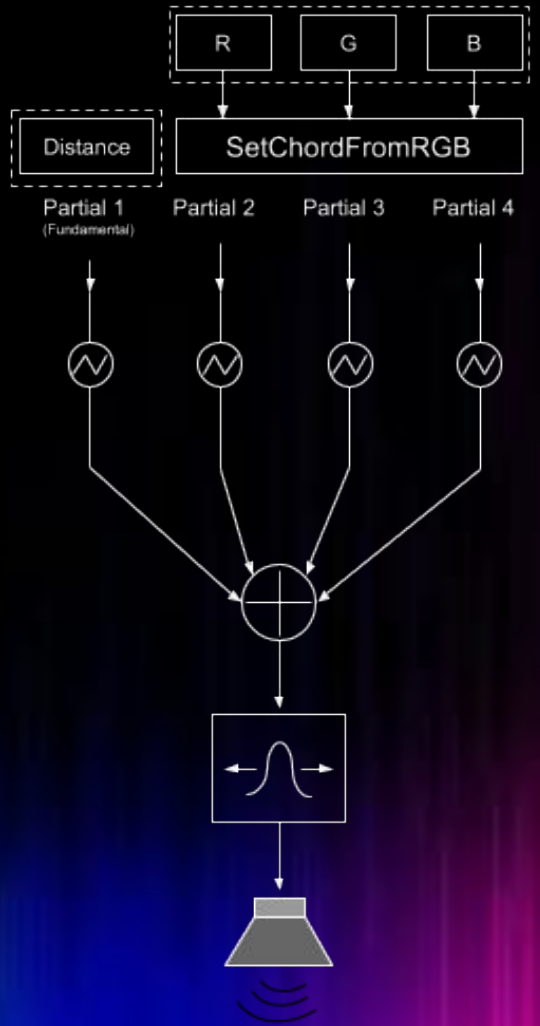


RGB based filter

- Final sound obtained by filtering the sum of four triangle wave oscillators with a bandpass filter
- Center frequency of the filter modulated by the RGB values
  - Fc = (b*80+g*200+r*800)/(r+g+b)
  - Blue colors sound darker, red colors sound brighter

Implementation

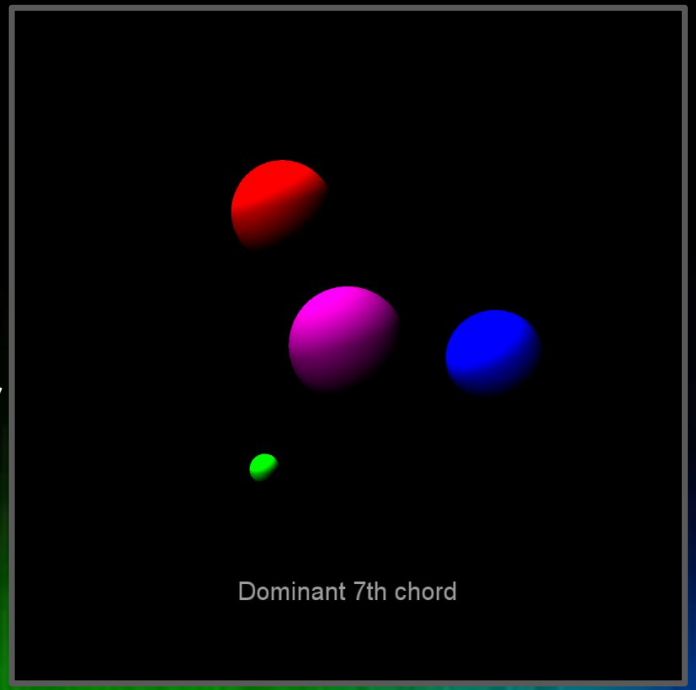
- All the musical system (aside from visual feedback) was developed directly on the Bela using C++ and the provided libraries
  - Oscillator library to generate the waveforms
  - Biquad library for the bandpass filtering
- RGB Color sensor is read by an AuxiliaryTask
- Distance sensor is read in the render() function, but only once every 60ms
  - Distance to pitch mapping is linear: [2, 100] cm -> [50, 2000] Hz



Graphical feedback

RGB and distance values sent to Processing

Color of central sphere: color sensed



Orbiting spheres size related to the sensed RGB values

The text tells the currently playing chord

Speed of orbitation related to the frequency obtained by the distance sensor

Thank you for your attention