

# BACODE READER



making sound by scanning barcode

20221124 Bohyun Choi

# Motivation

When scanning barcodes, a distinct beeping sound is produced. It's interesting to consider the possibility of transforming a **barcode scanner** into an instrument.

Perhaps we could generate a unique sound by converting images into barcodes. This could lead to some exciting possibilities!



# Settings



Each line of the barcode is spaced differently.

If the line is **dense**, the **pitch** of the sound is set to increase,  
and if the top and bottom are **long**, the **volume** of the sound is set to increase.

=> using Linear Regression

# Settings

```
color mousecol = get(mouseX, mouseY); //get current mousepoint's color val  
color mousecol_left = get(mouseX-5, mouseY);  
color mousecol_right = get(mouseX+5, mouseY);  
color mousecol_up = get(mouseX, mouseY-30);  
color mousecol_down = get(mouseX, mouseY+30);  
  
left_width = (red(mousecol)+green(mousecol)+blue(mousecol))  
-(red(mousecol_left)+green(mousecol_left)+blue(mousecol_left));  
  
right_width = (red(mousecol)+green(mousecol)+blue(mousecol))  
-(red(mousecol_right)+green(mousecol_right)+blue(mousecol_right));  
  
up_width = (red(mousecol)+green(mousecol)+blue(mousecol))  
-(red(mousecol_up)+green(mousecol_up)+blue(mousecol_up));  
  
down_width = (red(mousecol)+green(mousecol)+blue(mousecol))  
-(red(mousecol_down)+green(mousecol_down)+blue(mousecol_down));
```

## 1.Judging the density of barcode

If the color difference between the current mouse pixel and the left and right and the top and bottom pixels is large, it is judged that the distance is large

## Using Wekinator and Linear-Regression

input(5) : left, right, up, down width,  
current mouse pixel's color

output(2) : pitch, amp

# Settings

```
point = mouseX/random(3,10);
pointSize = width/point;

translate(pointSize/2,pointSize /2);

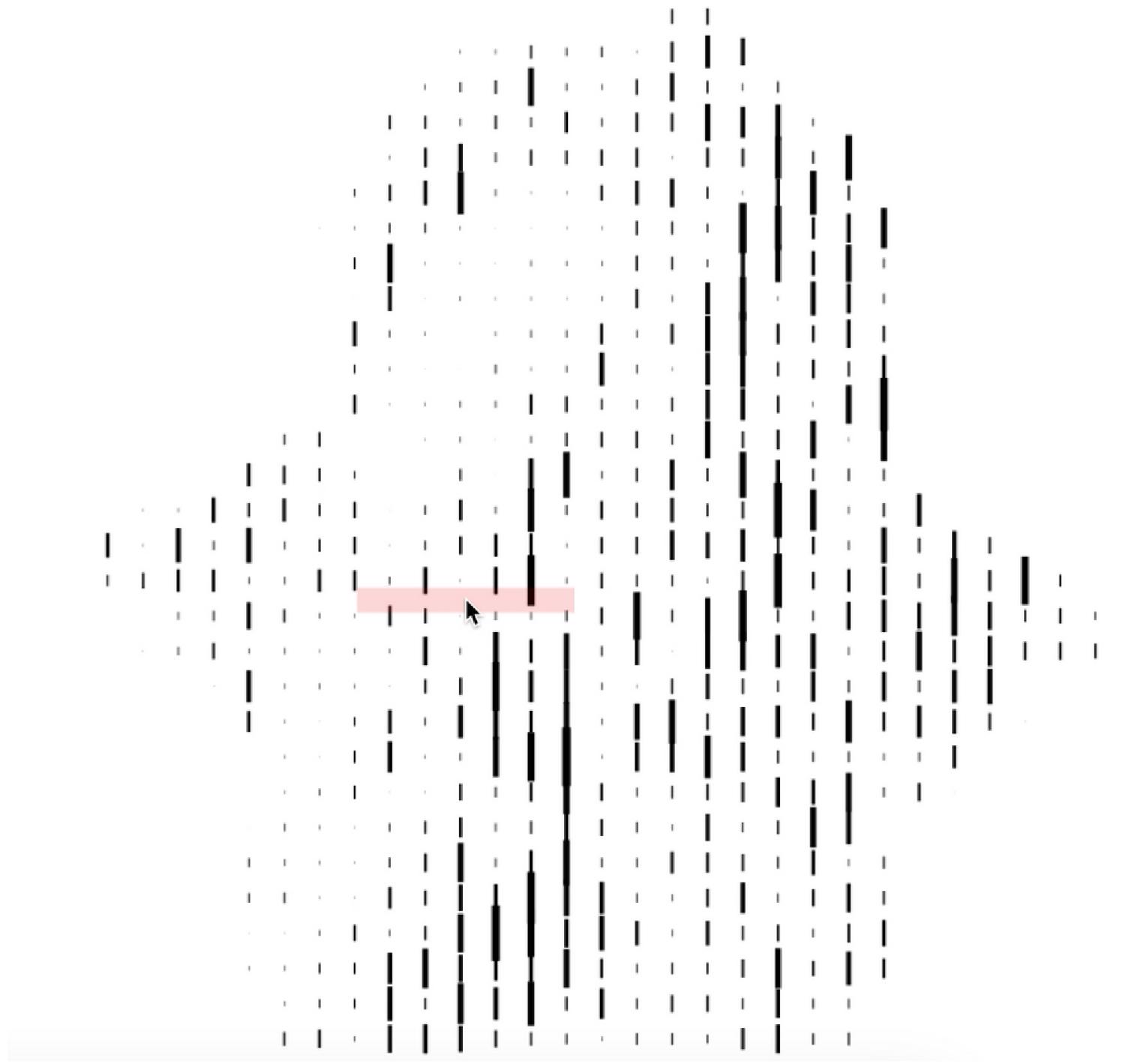
for (int i=0;i<point;i++)
{
  for (int j=0;j<point;j++)
  {
    c = img_load.get(int(i*pointSize),int(j*pointSize)); //return image color val by array
    width_size = map(brightness(c),0,255,random(1,6),0);
    rect(i*pointSize,j*pointSize,width_size,width_size*7);
  }
}

void keyPressed() {
  if (key == ' ') {
    println("Making barcode stopped");
    saveFrame("Canvas.png");
    //use saveFrame to save current Canva's graphics
    CanvasImage = loadImage("Canvas.png");
    barcode = !barcode;
  }
}
```

## 2.Convert image to BARCODE

The barcode are formed as densely the user's mouse goes to the right. If user press the space bar, user can stop with a barcode at a certain point in time.





Click/Drag mouse to scan your barcode