Brain Neurons

Generative Art - Task 3

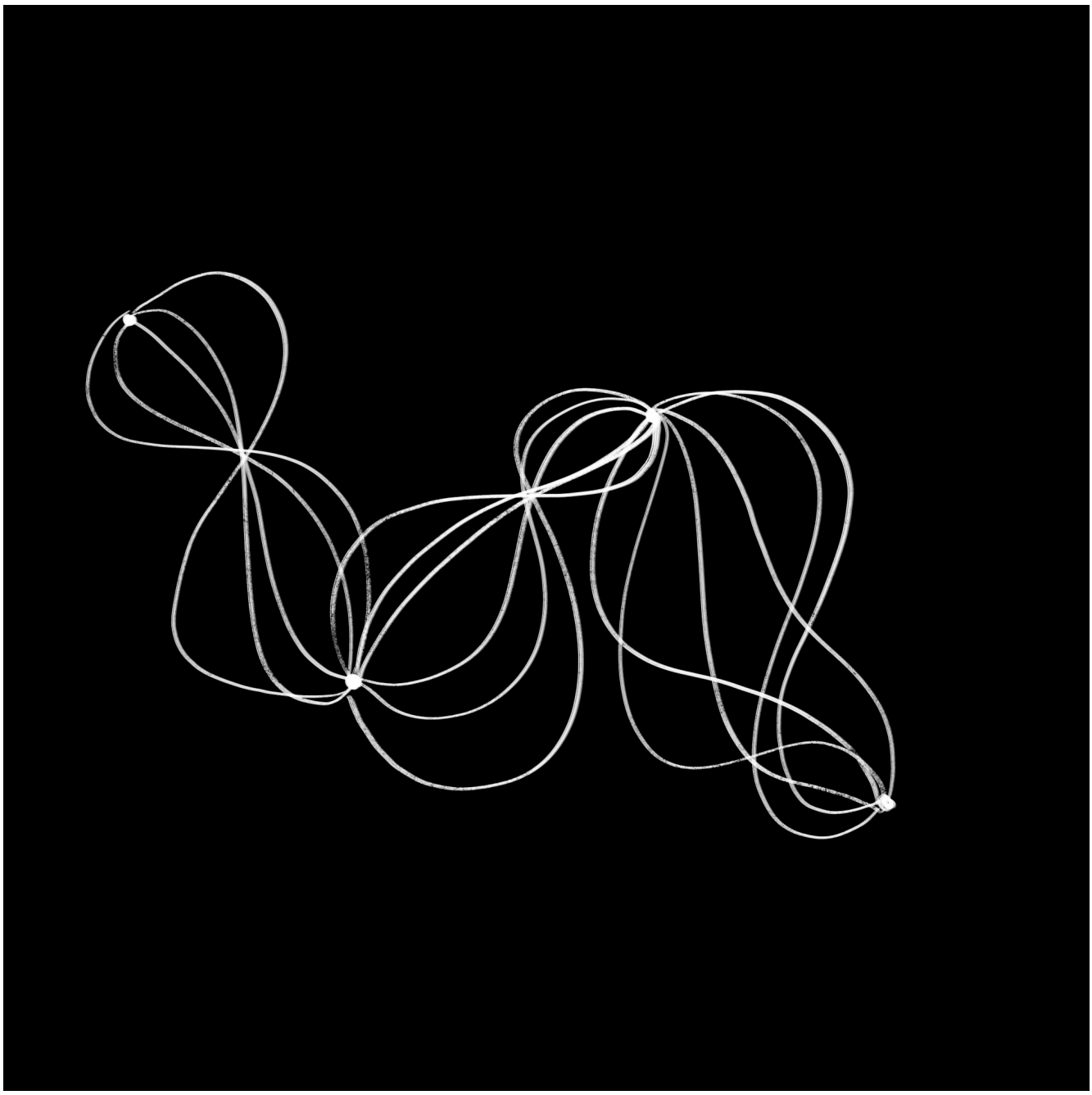
Will Ferrante

Project Idea

“Brain Neurons” is an interactive, generative art banner created using p5.js that visually simulates a neural network or thought process through flowing, interconnected lines. The system dynamically responds to mouse movement, generating animated parametric curves between tracked points to resemble neuron-like connections.

The effect represents the idea of thinking in motion - as you move the mouse, the screen builds a web of glowing, living connections, mimicking how thoughts and ideas form and branch out in the brain.

Design



Real-time Input

The banner tracks the user's Mouse Movement. It tracks the user's cursor in real-time, and stores the path in an array. The mouse position is taken every millisecond.

```
let trials = [];
```

*Initialising the array **trials***

```
// Track mouse movement
trials.push({x: mouseX, y: mouseY, t: millis()});

if (trials.length > 100){
  trials.shift();
}
```

inputting the data in the array using the 'push'. Data pushed is the mouse position (X, Y Co-ordinates) and the time (t)

If array reaches 100 enteries, the first ones are deleted and everything is shifted forwards

Techniques Used

Randomness was used for the generating the colour of the connected lines.

```
// Random colour shades
let r = random(100, 255);
let g = random(100, 255);
let b = random(100, 255);
stroke(r, g, b, alpha);
noFill();
```

getting random values for r,g,b which in turn return a random colour when put together as RGB Values

Noise was used to create movement in the background. The noise created erratic dots in the background.

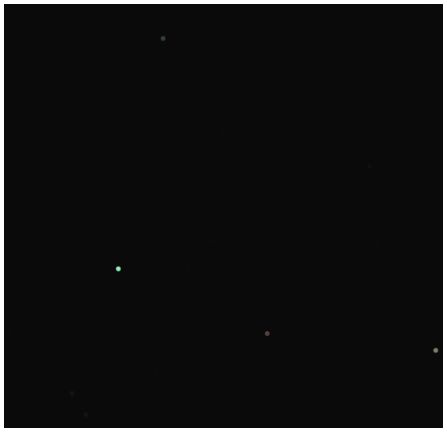
```
 // Set the noise level and scale.
let noiseLevel = 500;
let noiseScale = 1000;

// Scale the input coordinate.
let nt = noiseScale * frameCount;

// Compute the noise values.
let x = noiseLevel * noise(nt);
let y = noiseLevel * noise(nt + 10000);

// Draw the point.
strokeWeight(1.5);
point(x, y);
```

creating the noise in the background



Sine & Cosine Functions: used to animate the curves to give them a neural, wave-like energy.

```
// Louder volume = higher curve
let controlX = (p1.x + p2.x) / 2 + sin(p1.t * 0.005 + j) * curveStrength;
let controlY = (p1.y + p2.y) / 2 + cos(p2.t * 0.005 + i) * curveStrength;

bezier(p1.x, p1.y, controlX, controlY, controlX, controlY, p2.x, p2.y);
```

Data Stream

A real-time data stream using the microphone is used in the project. The project creates the arc heights depending on the noise level - the louder the noise, the higher the arc level.

```
// Start the mic
mic = new p5.AudioIn();
mic.start();

amplitude = new p5.Amplitude();
amplitude.setInput(mic);
```

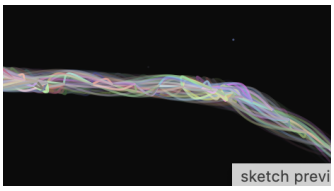
Starting the mic

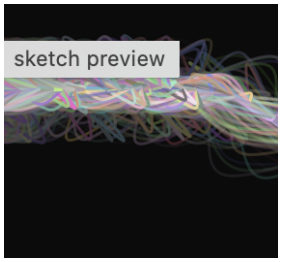
getting the volume levels

```
// Get volume level
let vol = amplitude.getLevel();
let curveStrength = map(vol, 0, 0.5, 10, 100); // scale it to use for arc height
```

```
+ sin(p1.t * 0.005 + j) * curveStrength;
+ cos(p2.t * 0.005 + i) * curveStrength;
```

the sine and cosine using the curveStrength set above

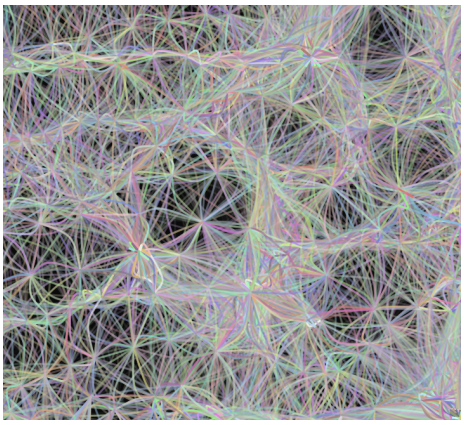




the curve before and after noise level was increased

Challenges Encountered

Since each frame compares every point in the mouse trail to every other point, the number of calculations grew rapidly, especially when the trail became longer. This was causing the art to lag and the screen to be cluttered, as more and more connections would be created and none deleted.



Addressing the Challenges

To address the above challenge, I limited the trail length to the last 100 mouse positions. This significantly reduced processing load while maintaining visual depth. By doing this, when the array reaches a 100 entries, the old ones are deleted and the remaining are shifted forwards.

```
if (trials.length > 100){
  trials.shift();
}
```

Adding an if statement to remove the old items in the array.

Critical Reflection

‘Brain Neurons’ reflects current trends in interactive, real-time generative media, where art is not static but shaped by user input and organic behaviors.

Application in the Media

This kind of generative artwork has strong potential in:

- Digital Design: As animated headers or ambient visuals for tech or neuroscience-related platforms.
- Interactive Installations: It can be projected in museums or exhibitions, reacting in real time to audience movement or touch.

Impact on Future Creative Industries

As tools like p5.js, TouchDesigner, and AI platforms become more accessible, artists can now design processes that evolve and adapt with interaction and data.

Brain Neurons, contributes to this evolution by offering a minimalist model of how structure and randomness can co-exist to form responsive digital systems. It emphasizes thought as movement, and creativity as connection—values that align with future-forward media environments.