

Artist Name: Ashley Harris
Title: Can You Beat the "Silence" ?
Medium: Generative Drawing and Sound Visualization
Size: 1500px x 1500px

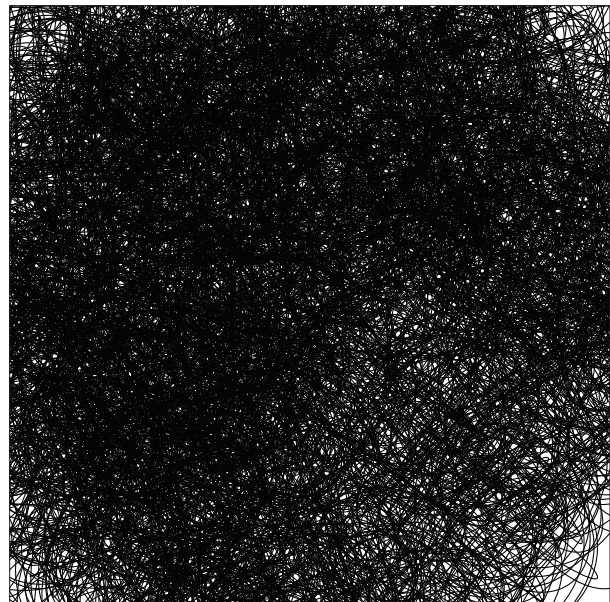
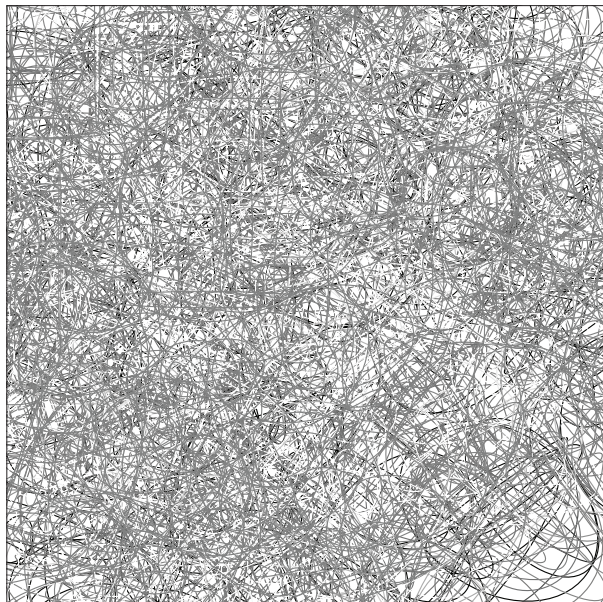
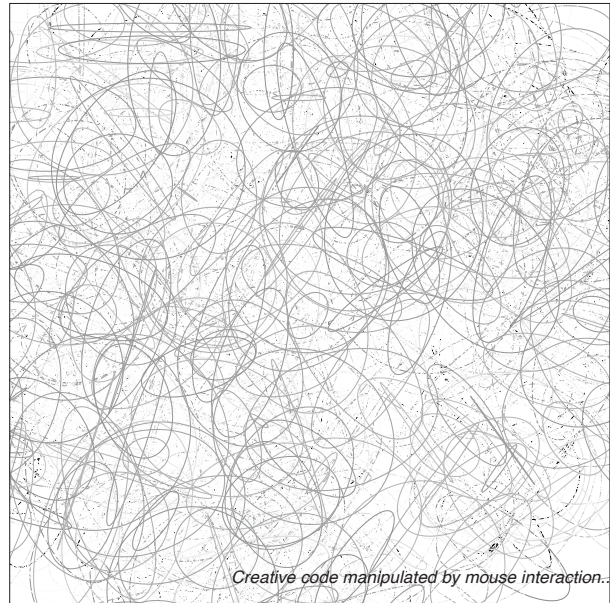
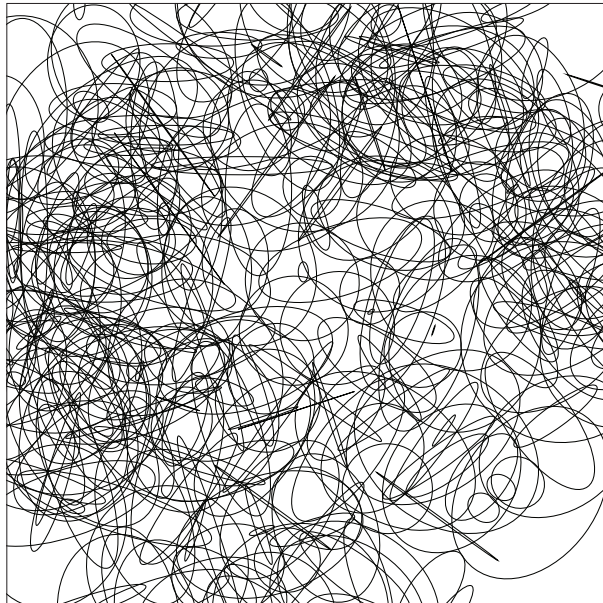
Artist Statement

In the modern era, what is silence? Silence is quiet? The absence of sound? The Human race is surrounded by sound, whether one believes it or not, in fact, if we were in total silence, it is theorized that one would not be able to handle it. Sound is engrained into the fabric of everyday living. But when does every day sound become sound pollution? How loud is the "silence"? We spend much of our lives unknowingly surrounded by sound. Can You Beat the "Silence" is an installation piece that requires the audience to test the very question it begs. Is passive sound louder than active sound? The piece is set up in a room and continuously picks up sound pollution as small white circles, as viewers enter the room, they are tasked with the challenge of returning the room to its original state of black, essentially beating the "silence" that haunts our everyday life. To do this, the viewers make purposeful noise such as claps, talking, yelling, etc. The louder the noise the bigger the circle and the darker the color. If the noise is loud enough, it contributes to beating the silence and returning the canvas to black. Can You Beat the "Silence" is created from the software program processing and explores the world of code-driven art and design. Through sound analysis via amplitude and frequency, the code picks up the sounds and delegates color and size to them depending on the volume.

Technical Statement

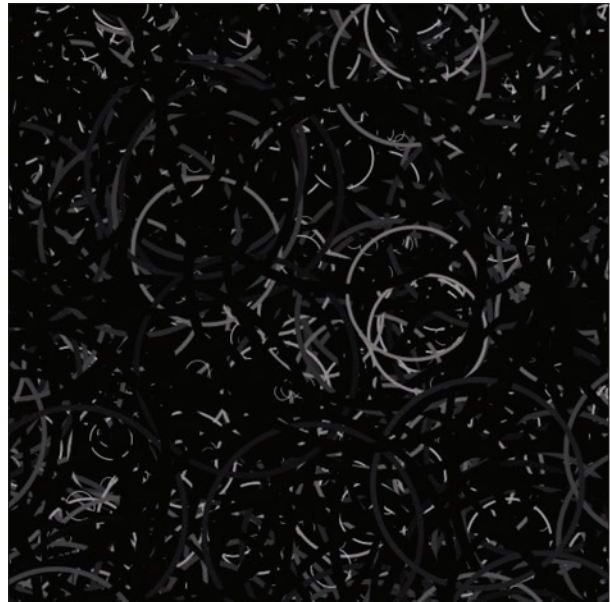
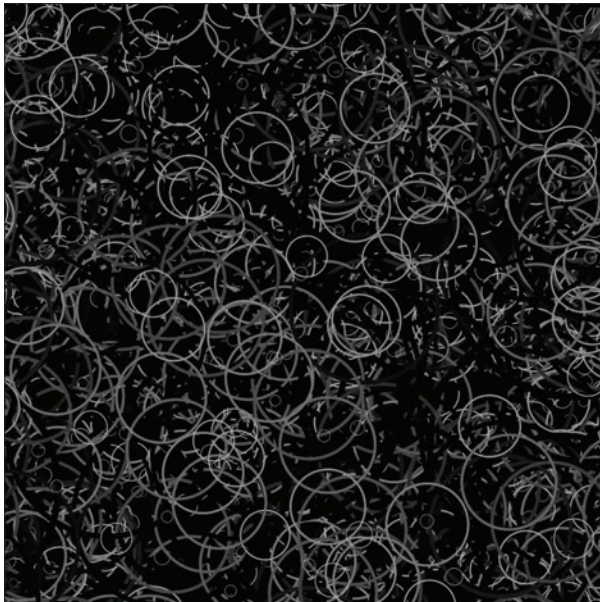
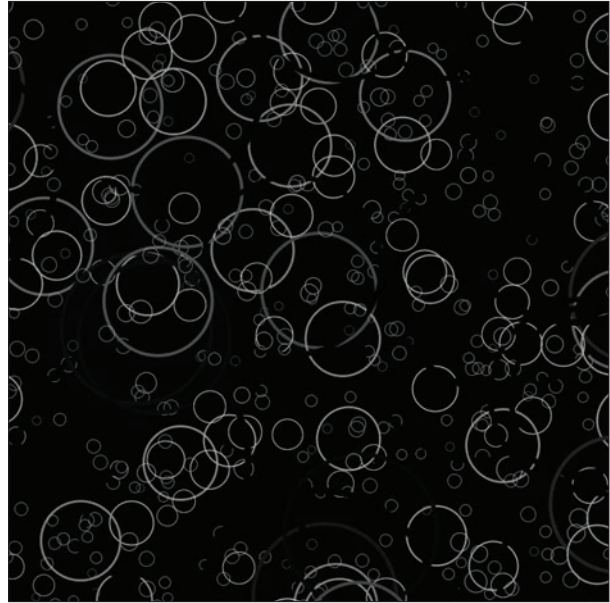
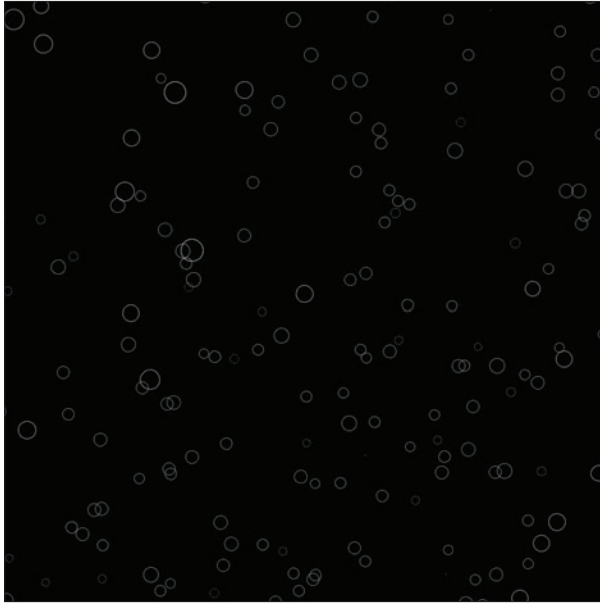
Can You Beat the "Silence"? was initially created as a visual document manipulated by a computer mouse within Processing. Processing is a base level programming language for artists and creatives, innovated by Casey Reas and Ben Fry. As time passed, I pulled to initial idea into the world of physical interaction and did this through the AudiInput sample in the Sound Processing Library created by The Processing Foundation. The AudiInput function allows for the code to pick up external sound through its speakers and create the images that it does.

1. Computational Drawings



2. Sound Visualization

Can You Beat the "Silence" ? Creative code that becomes a race against noise pollution.



```
import processing.pdf.*;

int timer=10;

import processing.sound.*;

Audioln input;
Amplitude loudness;

void setup(){
  beginRecord(PDF,"02.20_art448_Unit2_TestIdea_Version2.pdf");
  size(1500,1500);

  //beginRecord(PDF,"02.20_art448_Unit2_TestIdea_Version2.pdf");
  background(0);

  input = new Audioln (this, 0);
  input.start();

  loudness = new Amplitude(this);

  loudness.input(input);
}
void draw()
{
  //background(0);

  float x;

  float inputLevel = map(mouseY, 0, height, 1.0, 10.0);
  input.amp(inputLevel);

  float volume = loudness.analyze();
```

```
int size = int(map(volume, 0, 0.5, 1, 350));

x = (int)(255 - (loudness.analyze() * 800));
stroke(x);
//float red =
//float stroke = (red);

strokeWeight(loudness.analyze()*20);

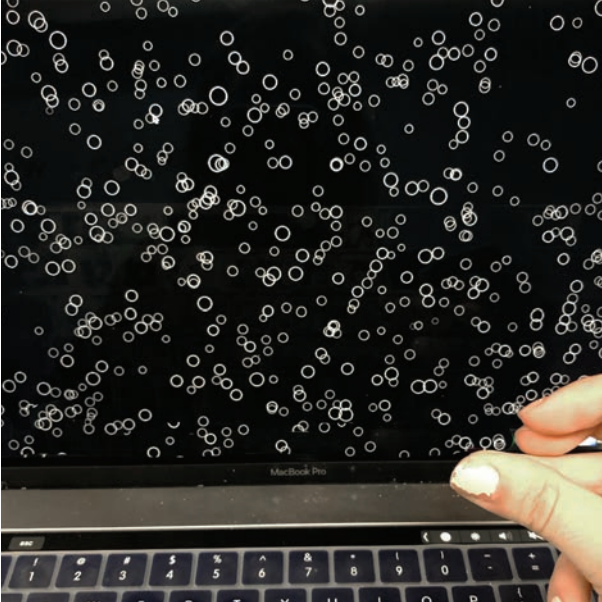
//timer=timer+1;
//for(int i=0; i<9; i=i+100)
{
  //translate(300,300);
  //fill(300,150,6);
  //smooth();
  //stroke(500,30,150);
}
//stroke(random(inputLevel, 900));
//for(int i=0; i<90; i=i+10)
//{
  noFill();
  //rotate(PI/5);
  //noFill();
  //smooth();
  random(size,size);
  ellipse(random(width), random(height), size, size);
  endRecord();
  if(timer==300);

  //}
  //if(timer==300)
  //{
  //endRecord();
  //}
  //}
```

Sequence of the computer code that creates the visual above

3. Physical Interaction

The more purposeful the sound, the thicker and darker the circle becomes.



4. Gallery Plan

An immersive space, reacting to the audience inside.

