

FLORES: PADRÃO DE PROCESSAMENTO DE IMAGEM FEITO NO PROCESSING

Lígia de Moura Miranda

Sistemas e Mídias Digitais – Universidade Federal do Ceará (UFC)
Fortaleza – CE – Brasil

***Abstract.** This article describes the process of creating an image processing algorithm made in Processing, with the purpose of honoring woman in a Computational Design course. This article describes the product named Flores, which was presented at a national art exhibition.*

***Resumo.** Este artigo descreve o processo de criação de um algoritmo de processamento de imagens feito em Processing, com a finalidade de homenagear mulheres em uma disciplina de Design Computacional. Este artigo descreve o produto nomeado Flores, o qual foi apresentado em uma exposição de arte nacional.*

1. Introdução

O projeto "Flores" é um algoritmo de processamento de imagens feito na cadeira de Design Computacional, no curso de Design na Universidade Federal do Ceará, no ano de 2016. Um exemplo de arte generativa feito em *Processing* que utiliza de programação para aplicar um padrão artístico em uma imagem pré-estabelecida.

A princípio, as inspirações e referências para criação deste código foram buscadas de forma despretensiosa, a partir de gostos pessoais e que nos agradassem visualmente, mas à medida que o código foi sendo criado veio a inspiração de tornar o projeto uma homenagem às mulheres da turma. O padrão de processamento funciona a partir da leitura de pixels em uma imagem em preto e branco onde cria ramos (fractais) e flores (círculos) de forma randômica, porém condicionados ao corte de tons de cinza da imagem.

O nome do projeto veio da associação da imagem visual ao ato de florescer e sobre este significado, nascer, se abrir e trazer a beleza e energia feminina. E assim homenageamos as mulheres que estavam ao nosso redor e que no dia a dia trabalharam conosco durante as aulas e momentos de contribuição para a criação do "Flores".

2. Arte Generativa e Processamento de Imagens

Segundo um dos pioneiros na área de arte generativa, Celestino Soddu¹, “arte generativa é a ideia realizada como código genético de eventos artificiais”, ele compreende a arte generativa em um paralelo com a natureza, onde os eventos variáveis do mundo criam formas e seres diversos. O artista, a partir da arte generativa, pode criar peças e imagens diferentes através das várias possibilidades que esse meio oferece, definindo assim uma forma abstrata de expressão e identidade artística. Ao longo dos anos, com o avanço da tecnologia, a arte generativa pôde ser experimentada em muitos meios, principalmente com o uso da programação.

Um processamento de imagens significa dar forma a uma imagem, onde esse processamento pode ser uma revelação de filme ou um processamento digital. Por definição, um processamento de imagem é “qualquer forma de processamento de dados no qual a entrada e saída são imagens tais como fotografias ou quadros de vídeo”². Toda área em que se possa utilizar de imagens para se obter algum resultado é uma área que irá utilizar um processamento de imagem, áreas como medicina, fotografia, vídeo, astronomia e obras de arte (SCURI, 2002). Em um processamento de imagens artístico, a programação nos dá a chance de criar a forma e expressão desejada dentro de uma imagem, podendo passar uma mensagem, seja ela direta ou abstrata.

3. Etapas de criação

Por ser um trabalho de uma cadeira, o processo de concepção das ideias foi feito em três etapas. Não havia especificamente orientações sobre que tipo de trabalho deveria ser feito, apenas que fosse um processador de imagens com padronização.

O trabalho foi feito em duplas, e nessa primeira parte trazíamos algumas ideias de imagens e até exemplos de instalações multimídia, projetos que de alguma forma envolviam a arte e a programação juntas.

¹ Informação retirada do site “GENERATIVE ART E GENERATIVE DESIGN”. Disponível em: <<https://www.soddu.it/>>.

² Informação retirada do site “PROCESSAMENTO de imagem. Wikipédia: a enciclopédia livre”. [São Francisco, CA: Fundação Wikimedia], 2021. Disponível em: <https://pt.wikipedia.org/wiki/Processamento_de_imagem>.

3.1. Brainstorm

O *brainstorm* foi feito com a participação exclusiva da equipe, composta por duas pessoas, Lígia de Moura Miranda e Giovanna Magda Sousa e Silva, onde foi utilizado o modelo de *brainstorm* não estruturado. Segundo SEBRAE (2005), nesse modelo os membros do grupo expõem suas ideias conforme elas surgem em suas mentes, sendo um método mais descontraído e possibilitando um ambiente mais relaxado. No geral, existem etapas básicas para todos os métodos de *brainstorm*, mas concentramos em três etapas essenciais: introdução, geração de ideias e definição. A primeira parte consistiu na introdução ao projeto, onde buscamos entender e pesquisar, principalmente no *Google* Imagens e no *Youtube*, exemplos de processamento de imagens usando programação, para assim entendermos como funciona e qual estilo de resultado geralmente é obtido. A segunda etapa focou na geração de ideias, onde buscamos inspirações para a criação do nosso processamento de imagem digital, nossa atenção foi direcionada às imagens que mais nos agradaram visualmente (FIGURA 1) e como elas poderiam ser usadas como inspiração (FIGURA 2). A busca das referências foi baseada principalmente nos nossos gostos pessoais e como gostaríamos que fosse o nosso processamento de imagens.

Na nossa terceira e última etapa de *brainstorm*, tivemos questionamentos sobre representatividade feminina no meio da programação. De acordo com o estudo do IBGE³, as mulheres são minoria em cursos de graduação nas áreas ligadas às ciências exatas e à esfera da produção, representadas por apenas 13,3% das matrículas nos cursos de Computação e Tecnologia. Isso faz as mulheres subestimarem a sua própria capacidade nesse ambiente e devido a essas questões, direcionamos o nosso trabalho a algo que pudesse de alguma forma homenagear as mulheres que estavam ao nosso redor, trazendo assim um sentido maior e mais profundo ao projeto.

Diante das várias imagens que tivemos acesso, decidimos usar nosso trabalho como uma homenagem às mulheres. Procuramos tudo aquilo que de alguma forma representava feminilidade, a fim de buscar um meio de colocar aquela arte dentro de um trabalho que envolvia um ambiente e assunto o qual estávamos nos aprofundando naquele momento.

³ Informação retirada do site “Agência IBGE Notícias”. Disponível em: <<https://agenciadenoticias.ibge.gov.br/agencia-sala-de-imprensa/2013-agencia-de-noticias/releases/30172-estatisticas-de-genero-ocupacao-das-mulheres-e-menor-em-lares-com-criancas-de-ate-tres-anos>>.

Figura 1. Exemplo das primeiras referências encontradas.

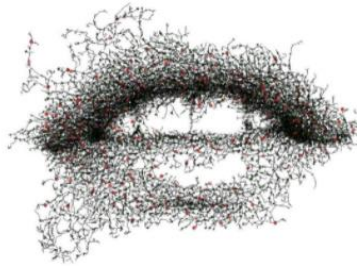


Figura 2. Referência principal para execução do trabalho.



Escolhemos como referências principais as imagens que continham o visual de ramos e flores, as quais metaforicamente se associavam ao nosso sentimento diante do trabalho, o de florescer em uma área pouco explorada por nós. As instalações de Daniel Rozin⁴ (FIGURA 3) nos inspiraram a utilizar imagens de pessoas reais na apresentação do trabalho, onde o objeto principal era aquele que utilizava o processamento. Apesar da imagem utilizada como oficial no projeto não ser a de uma pessoa real, produzimos um vídeo que tinha uma imagem, dentro do processamento generativo, de cada mulher presente na turma, a fim de homenagear cada uma delas.

Figura 3. Instalação multimídia de Daniel Rozin.



⁴ Informação retirada do site “DANIEL ROZIN INTERACTIVE ART”. Disponível em: <<http://www.smoothware.com/danny/>>.

3.2. Pesquisa de Código

Após ter definido nossas inspirações, passamos para a segunda etapa que consistia em procurar formas de criar o processamento. Nessa pesquisa, utilizamos de sites a fóruns abertos que tivessem algo que se aproximasse da arte definida.

Dentre os vários sites, alguns deles foram o site oficial do *Processing*, a biblioteca aberta *OpenProcessing* (FIGURA 4) e o livro online “*Nature of Code*” de Daniel Shiffman, o último em especial sendo de muita importância para o padrão de ramos do processamento.

Para a criação dos ramos, utilizamos a ideia dos fractais. No livro “*Nature of Code*”, o capítulo 8 (FIGURA 5) ensinava como criá-los e toda a lógica matemática por trás do que é um fractal. As “flores” foram criadas a partir de pontos com coloração vermelha, que no conjunto da obra formavam os ramos e as flores.

Figura 4. OpenProcessing, uma biblioteca de códigos abertos com utilização livre.

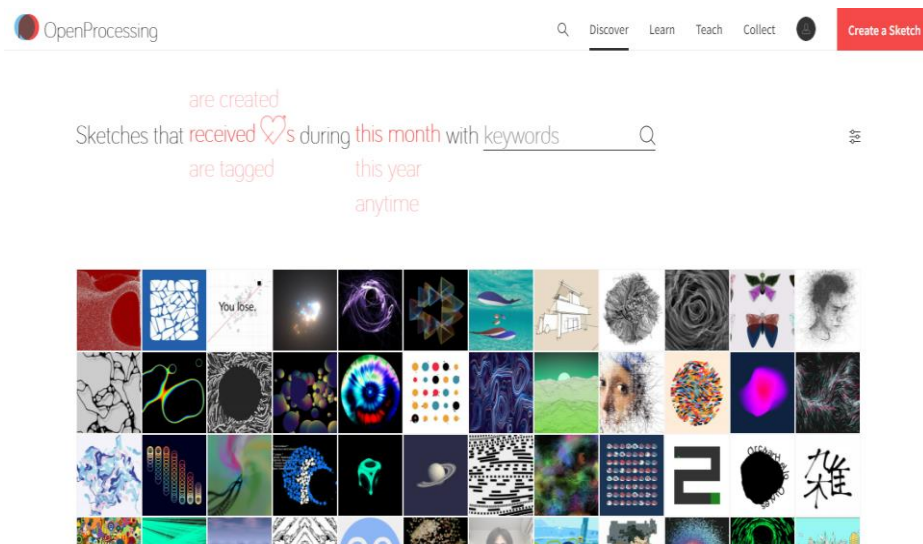
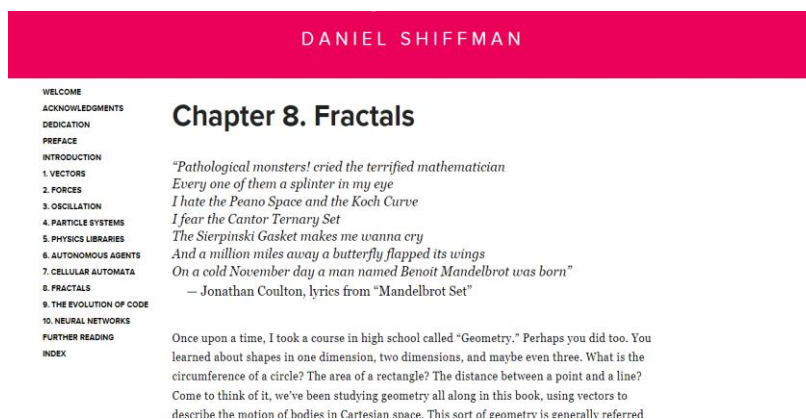


Figura 5. Livro online “*Nature of Code*”, de Daniel Shiffman.



3.3. Construção do código

O *Processing* é uma ferramenta gratuita voltada a codificação de artes visuais baseada na linguagem Java. Essa ferramenta é amplamente usada por designers e artistas para criação de seus trabalhos, como também estudantes que estão iniciando seus conhecimentos na programação. No “Flores”, além do *Processing*, também é utilizada a biblioteca “p5.js” que é baseada em *Javascript*, também gratuita e com as mesmas finalidades que o outro software.

Após a pesquisa de código, demos início a criação do nosso projeto. Para criar esse processamento, independente do padrão escolhido, é preciso fazer uma leitura da imagem que será utilizada, guardando a informação de cada pixel. No nosso código, o processamento é feito se baseando nos tons de cinza, então obrigatoriamente a imagem precisa ser em preto e branco.

No primeiro momento, determinamos o básico do código, como o tamanho da tela e variáveis (FIGURA 6). As imagens que serão utilizadas precisam ser vinculadas no código e estar presente numa pasta junto aos arquivos do *Processing*. Essas imagens são carregadas dentro do *software* e é no próprio código que selecionamos a imagem que irá passar pelo processamento.

Figura 6. Determinação do tamanho da tela e variáveis que serão utilizadas.

```
aggregation_web aggregation classAggregation classAggregationPoint classTarget lookForTarget ▼
float [] [] alvoAuxiliar;
PImage [] images;
boolean showImage = false;
boolean record;

void setup ()
{
  size (900, 700);
  smooth();

  frameRate (300);

  images = new PImage [7];
  images[0] = loadImage ("lirja.png");
  images[1] =loadImage ("blackwoman2.jpg");
  images[2] =loadImage ("anal.jpg");
  images[3] =loadImage ("hosana1.png");
  images[4] =loadImage ("oi.png");
  images[5] =loadImage ("titulo4.png");
  images[6] =loadImage ("obrigada.png");
  input = images [imageIndex];

  initBranches(alvoAuxiliar, input);
}
```

Uma imagem digital é feita de *pixels* e cada *pixel* pode armazenar várias informações. Para torná-los um ponto alvo a ser considerado dentro do processamento, armazenamos informações de coordenada, tom de cinza e tamanho (FIGURA 7). Dessa forma, poderemos

fazer a verificação de cada *pixel* da imagem (FIGURA 8), a fim de encontrar os pontos que servirão de base para a arte a ser criada.

Figura 7. Recorte de tom de cinza.

```
aggregation_web | aggregation | classAggregation | classAggregationPoint | classTarget | lookForTarget ▼
// verifica quais pontos são válidos para se tornarem "alvos"
PVector pontoValido (int [] cinza, int x, int y, int W, int H, int a)
{
    PVector pos = new PVector (0, 0);
    int index = y*W+x;
    color c = cinza [index];

    if (a == 15 || ehPontoValido (brightness (c)))
    {
        pos.x = x;
        pos.y = y;
    }
    else
    {
        pos = pontoValido (cinza, (int) random (W), (int) random (H), W, H, a++);
    }

    return pos;
}
```

Figura 8. Validação de pontos dentro imagem.

```
aggregation_web | aggregation | classAggregation | classAggregationPoint | classTarget | lookForTarget ▼
// armazena as cordenadas dos pontos válidos para serem desenhados
float [] [] totalPontosValidos (int n, PImage img)
{
    img.loadPixels();
    float [] [] pontoSelecioneado = new float [n] [2];

    PVector pos;
    for (int i = 0; i < pontoSelecioneado.length; i++)
    {
        pos = pontoValido (img.pixels, (int) random (img.width), (int) random (img.height), img.width,

        pontoSelecioneado [i] [0] = pos.x;
        pontoSelecioneado [i] [1] = pos.y;
    }

    return pontoSelecioneado;
}
```

Após a varredura em todos os *pixels* da imagem, seleciona-se quais irão se tornar pontos a serem desenhados com a condicional de brilho e escala de cinza, e assim o código inicia seu processamento de ramos e flores (FIGURA 9). Por causa das condições estabelecidas, a imagem processada consegue fazer essa variação de brilho onde a parte da arte que mais contém o tom de preto, se acumula os ramos e flores, e no local onde os pixels estão mais próximos do branco,

não é desenhado nenhuma arte. Portanto, pelo contraste e efeito de luz e sombra, conseguimos ver a mesma imagem com a variação de brilho que ela tem originalmente.

Figura 9. Código de como serão desenhados os ramos e as flores.

```
aggregation_web aggregation classAggregation classAggregationPoint classTarget lookForTarget ▼
'/ desenha as linhas dos ramos e as rosinhas
void draw ()
{
  stroke (0, 150);
  strokeWeight (sw);
  if (filhote != null && filhote.size() > 0)
  {
    if (filhote.size() >= 2 && filhote.size() <= 3)
    {
      for (int i = 0; i < filhote.size(); i = i+2)
      {
        noStroke();
        fill (255, 0, 0, 80);
        ellipse (x+i, y+i, sw*2.3, sw*2.3);
        noFill();
        stroke (0, 80);
      }
    }
    for (int i = 0; i < filhote.size(); i++)
    {
      line (x, y, filhote.get(i).x, filhote.get(i).y);

      filhote.get(i).draw();
    }
  }
}
```

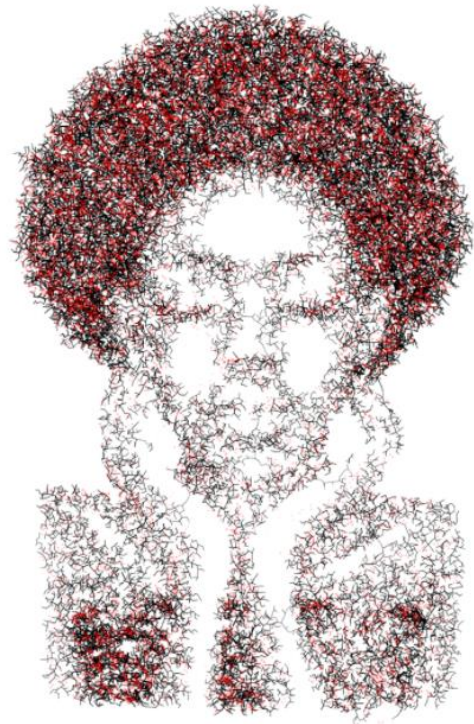
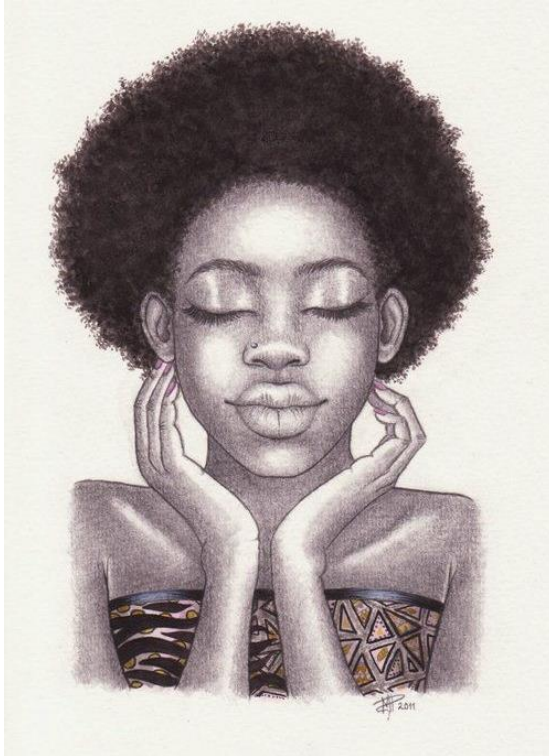
4. Flores

4.1. Versão final

A imagem escolhida para representar (FIGURA 10a), além de ter sido selecionada com base nas referências pesquisadas, também nos passava a ideia de força e feminilidade. E após a criação do código, o processamento de imagens foi feito (FIGURA 10b).

Também foi aplicado o processamento em fotos das alunas da cadeira e depois todas as imagens foram compiladas em um vídeo com a descrição da poesia “A caça” de Adna Rahmeier, que fala sobre as qualidades das mulheres e o poder feminino. Dessa forma, foi possível homenagear as integrantes da turma, além de nós mesmas, após ter finalizado toda a criação do código.

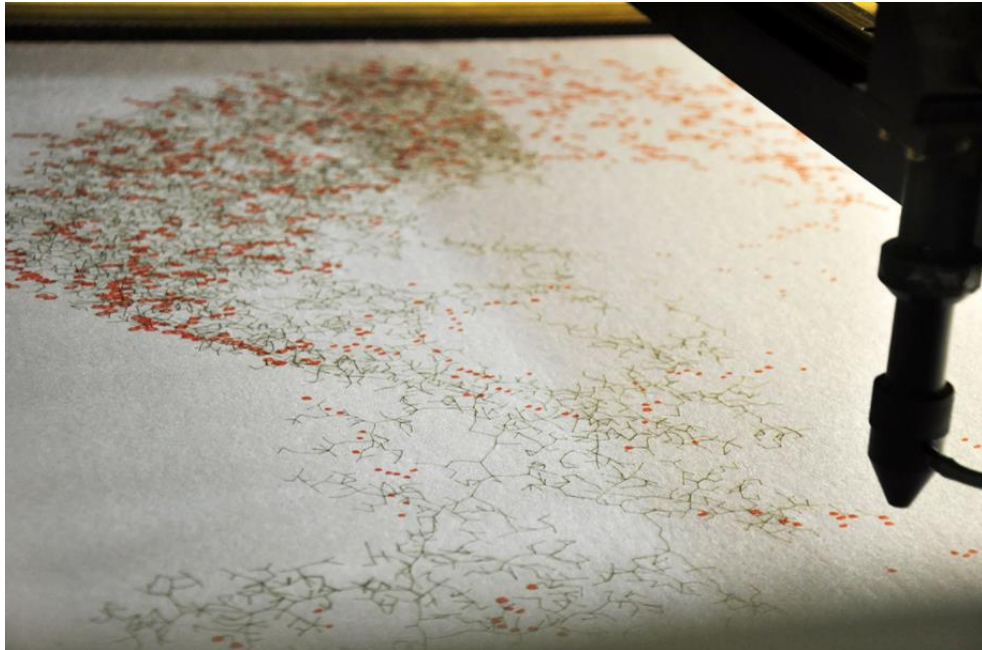
Figura 10. A) Imagem utilizada para aplicar o processamento de imagem “Flores” b) Mesma imagem após o processamento.



4.2. Produção de arte física

Ao final do trabalho, além da imagem processada e do vídeo, também foi feita uma aplicação em placa de mdf (FIGURA 12) com a imagem oficial marcada a *laser*. Para a aplicação, utilizamos também código de programação no próprio arquivo de *Processing*, onde fomos capazes de salvar o arquivo em modelo DXF da aplicação dos ramos e flores separadamente. Todo o processo de aplicação levou 2 dias, devido ao tamanho da placa e detalhamento da arte. Os ramos da arte foram feitos a *laser* e as flores, feito na mesma máquina, porém utilizando uma caneta de cor vermelha.

Figura 12. Imagem da aplicação em placa de MDF.



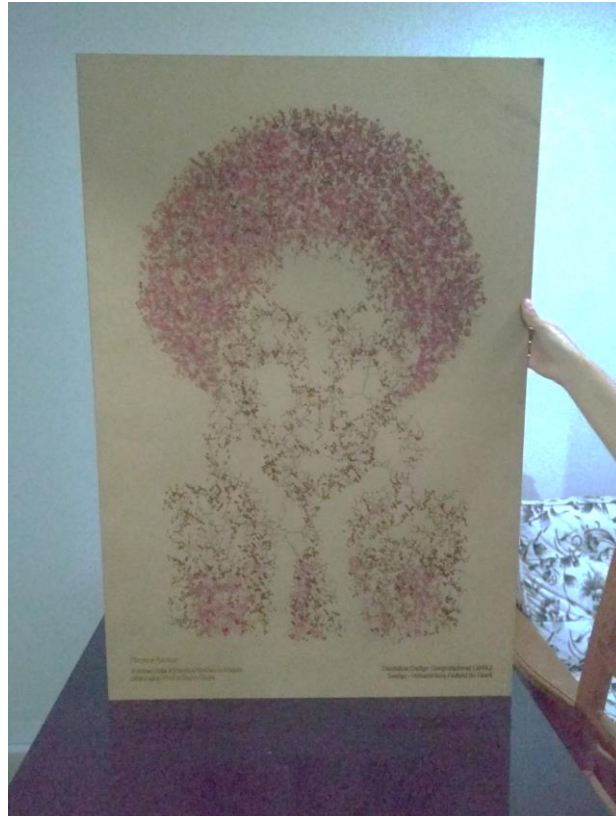
4.3. Exposição

Até a primeira criação de arte física, ainda estávamos trabalhando dentro do que a cadeira pedia, porém devido a boa execução do trabalho decidimos aplicar a nossa arte física para a VI Mostra de Jovens Designers. Selecionado no ano de 2018 dentre 403 inscritos de todo o Brasil, nosso projeto foi levado até São Paulo, ficando exposto por quase dois meses junto a outros 37 projetos que também foram selecionados.

Para a exposição, criamos outro exemplar da mesma arte (FIGURA 13), novo e em melhor qualidade. Não foi possível a equipe estar presente na exposição, mas o nome dos integrantes, o da instituição e do nosso estado foi levado a essa mostra, dentre vários trabalhos artísticos e de design de produto.

A mostra selecionou um dos projetos para ser levado a Milão, na Itália, infelizmente o nosso não foi selecionado, porém conseguir levar esse projeto a esse nível de qualidade já foi uma grande conquista.

Figura 13. Segunda aplicação em placa de MDF.



5. Conclusão

A experiência de criar o projeto “Flores” foi engrandecedor e trouxe para mim uma nova perspectiva de uma área, de um assunto o qual eu havia por muito tempo ignorado ou até mesmo achado que não tinha capacidade para tal. Perceber que programação e os conceitos de lógica é apenas uma forma diferente de ver o mundo e como as coisas funcionam, e através da ferramenta do *Processing*, um software acessível, gratuito e com uma vasta biblioteca de códigos, fui capaz de criar arte e design. Através dessas percepções, também se abriu a reflexão sobre representatividade feminina no meio dos sistemas, o qual foi o foco da homenagem desse trabalho artístico.

Devido ao êxito do trabalho, o próximo passo é aperfeiçoar o projeto, disponibilizando o código em fóruns e tornando-o mais acessível e prático. Há também o desejo de fazer o que chamamos de instalação multimídia, onde seria capaz de aumentar o alcance dessa experiência e dessa forma criar artes mais pessoais, onde de fato o objeto principal da arte é a pessoa que

utiliza. Desejo que esse meio cresça mais e que tenhamos mais projetos como este, sendo representados também por mulheres no meio da programação e do design.

6. Referências

AGÊNCIA IBGE NOTÍCIAS, 2021. Disponível em:

<<https://agenciadenoticias.ibge.gov.br/>>. Acesso em: 02/02/2022.

DANIEL ROZIN INTERACTIVE ART". Disponível em:

<<http://www.smoothware.com/danny/>>. Acesso em: 02/02/2022.

DIVISÃO DE DEPARTAMENTO DE IMAGENS. Disponível em:

<<http://www.dpi.inpe.br/>>. Acesso em: 02/02/2022.

GENERATIVE ART. Disponível em: <<http://generativeart.com/>>. Acesso em: 02/02/2022.

GENERATIVE ART E GENERATIVE DESIGN. Disponível em:

<<https://www.soddu.it/>>. Acesso em: 02/02/2022.

OPENPROCESSING. OpenProcessing, 2021. Página inicial. Disponível em:

<<https://openprocessing.org/>>. Acesso em: 02/02/2022.

PROCESSAMENTO de imagem. Wikipédia: a enciclopédia livre. [São Francisco, CA: Fundação Wikimedia], 2021. Disponível em:

<https://pt.wikipedia.org/wiki/Processamento_de_imagem>. Acesso em: 02/02/2022.

PROCESSING. Processing, 2021. Página inicial. Disponível em: <<https://processing.org/>>.

Acesso em: 02/02/2022.

SCURI, Antonio. Fundamentos da Imagem Digital. Rio de Janeiro, 2002. Disponível em:

<<http://www.inf.ufes.br/~thomas/graphics/www/apostilas/CIV2801ScuriImgDigital.pdf>>.

Acesso em: 02/02/2022.

SEBRAE. Manual de Ferramentas da Qualidade, 2005. Disponível em:

<https://edisciplinas.usp.br/pluginfile.php/2621303/mod_folder/content/0/SEBRAE%202005.pdf?forcedownload=1>. Acesso em: 02/02/2022.

SHIFFMAN, Daniel. Nature of Code, chapter 8. Califórnia, 2012. Disponível em:

<<https://natureofcode.com/book/chapter-8-fractals/>>. Acesso em: 02/02/2022.