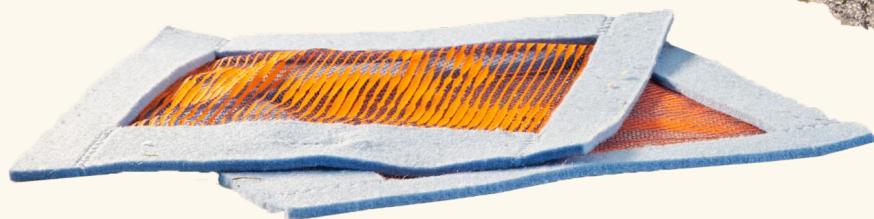


# CARRYING MEMORIES

Digital Craftsmanship Q4 23/24

**Group 9**  
Sophie van der Burg - 2054388  
Max Nielsen - 2102714  
Luke Smit - 1826980  
Jasmijn Vugts - 1450034



## ITERATION 1

Magic Machine Workshop

Devision in the team over our chosen value

# Orderliness



A backback with all the tools you might need during whatever adventures the day brings you



Inspired by dora's backpack



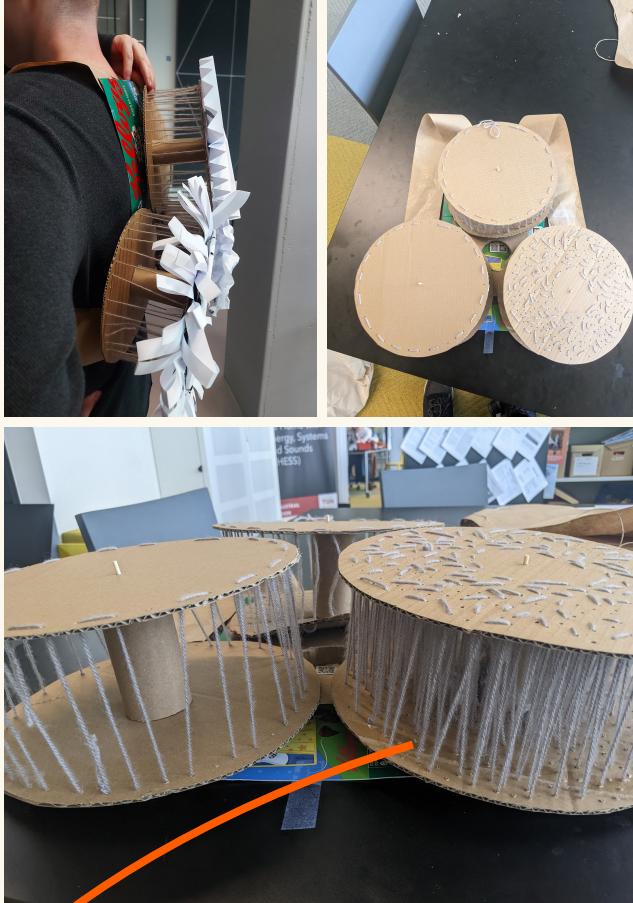
Completely opens up to have easy acces to all tools



## ITERATION 2

Magic Machine Iteration

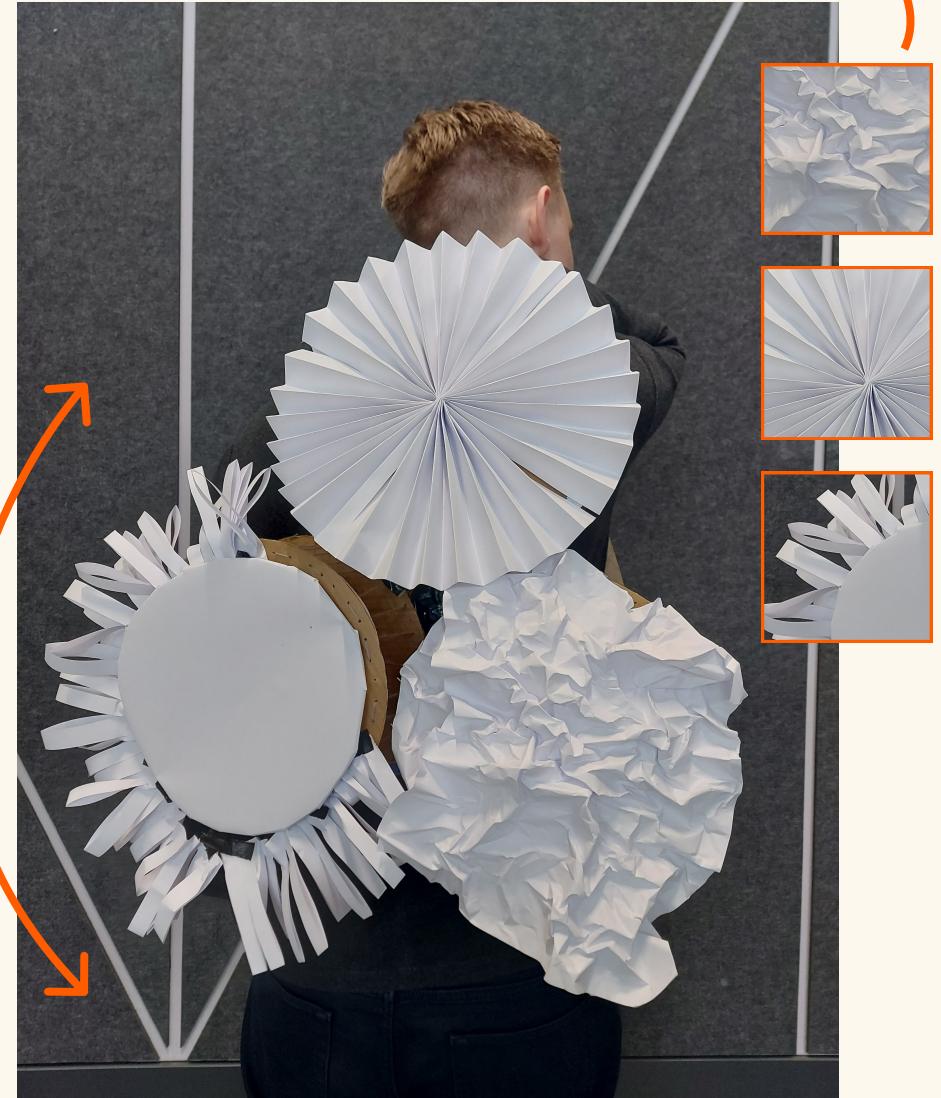
Creating the bag in context: How would the bag be packed and how would it be used during the day?



Elastic bands to simply stick in whatever you need

To make packing the bag more spontaneous

Textures to feel what pocket you are reaching for so you don't have to take off the bag during the adventures



If you want to be prepared for anything, you need a **HUGE** bag to take all your tools with you...

By becoming ultra independent, we isolate ourselves from others.

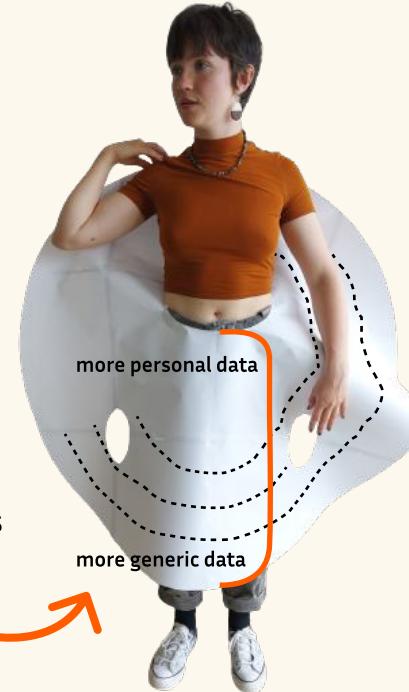
## ITERATION 3

Barry the skirt

The conclusion of 2nd iteration found us exploring spontaneous interaction with others through sharing, instead of isolating yourself with your preparedness.



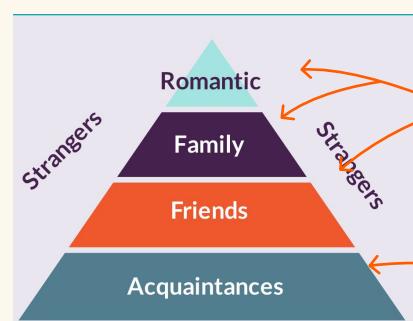
Layers/rings of connection, deepest layer of connection is closest to yourself



It can also act as a chair!

Experimenting with showing/hiding your body and personal data based on your current feeling.

Sharing a variety of data of yourself with a variety of other people to strengthen connections.



*The spectrum of connection*  
with who would you share what type of data?

## ITERATION 3.5

More experimentation with showing and hiding





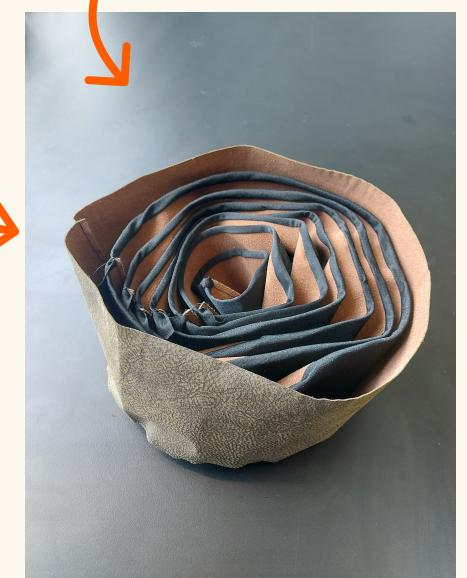
## ITERATION 4

*Tree of Memories*

*Instead of sharing general, abstract data of yourself, we share memories or experiences: Something more tangible for others to understand.*

You can release the bag fully or partially depending on what or how much you want to show.

The ability to squish it or leave it in its natural shape



The deeper the memory in the bag, the more complex the memory is to show to yourself or others.

What kind of memories, who these memories are for, is still unanswered

## FINAL CONCEPT

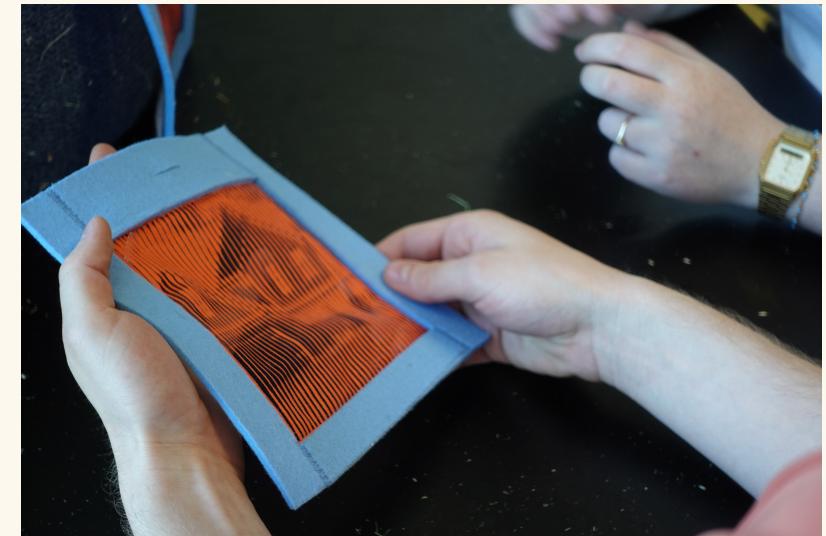
Our previous iteration consisted of a wide variety of interesting elements. However, it needed a more specified narrative to deepen the elements.

Options considered were:

- Parent/child memories
- Memories of the people we lost
- Reflecting bag for your high school
- Curriculum Vitae collected through uni

Eventually we chose **grief** because we were drawn by the complexity of the narrative. Thereby, how grief intertwines with interpersonal relationships links to our findings in the previous iterations.

Each memory representation visualization is a memory that you hold with your lost one.



Remove the memories from the bag to personally reflect or closely share with others.



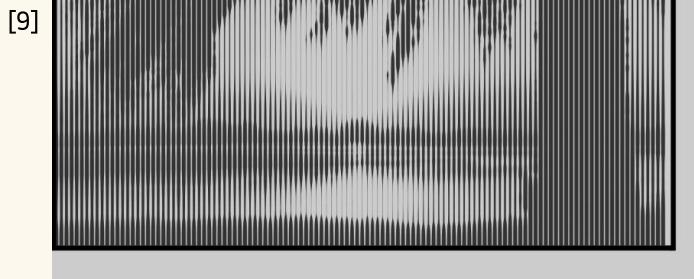
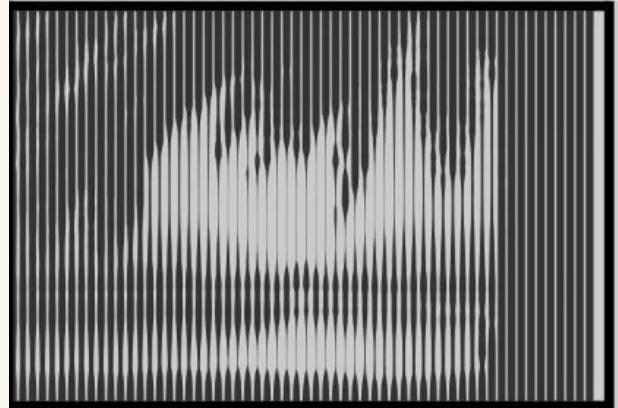
A bag that holds memories of a person you were very close with (relative, partner, friend).

Filled with a bio degradable material. The loose knit, enables the bag to shrink over time—together with the grief.

TO GET HERE WE DID SOME EXPLORATIONS....

# CODE EXPLORATION

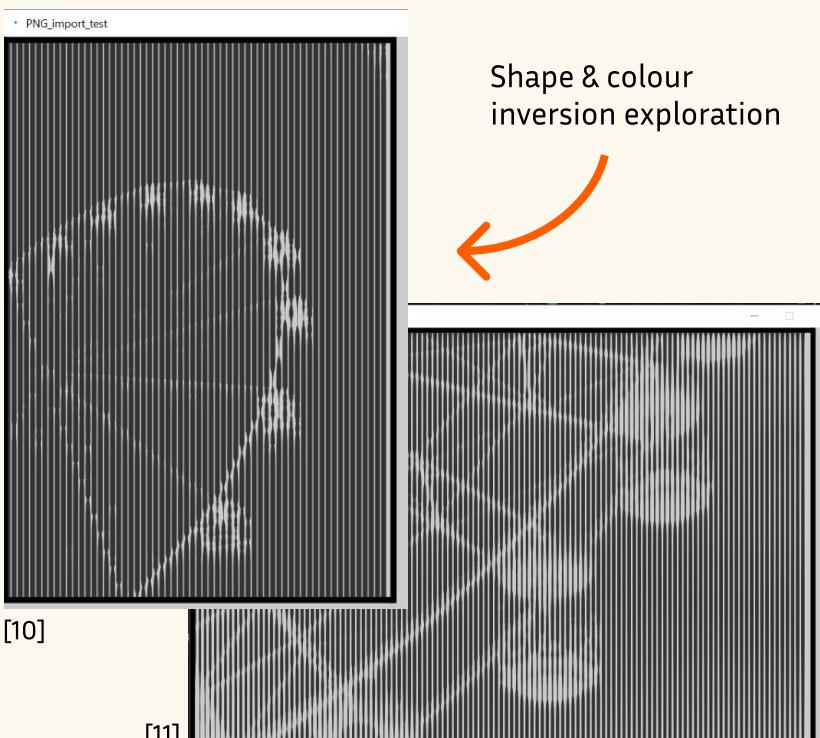
*Different line explorations*



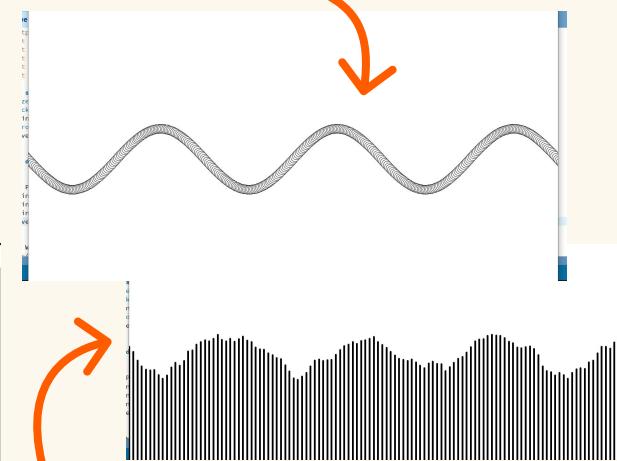
Initial tests for abstracting the memories

Wholly different data concept: algae in Canadian lakes

Wave visualisation exploration



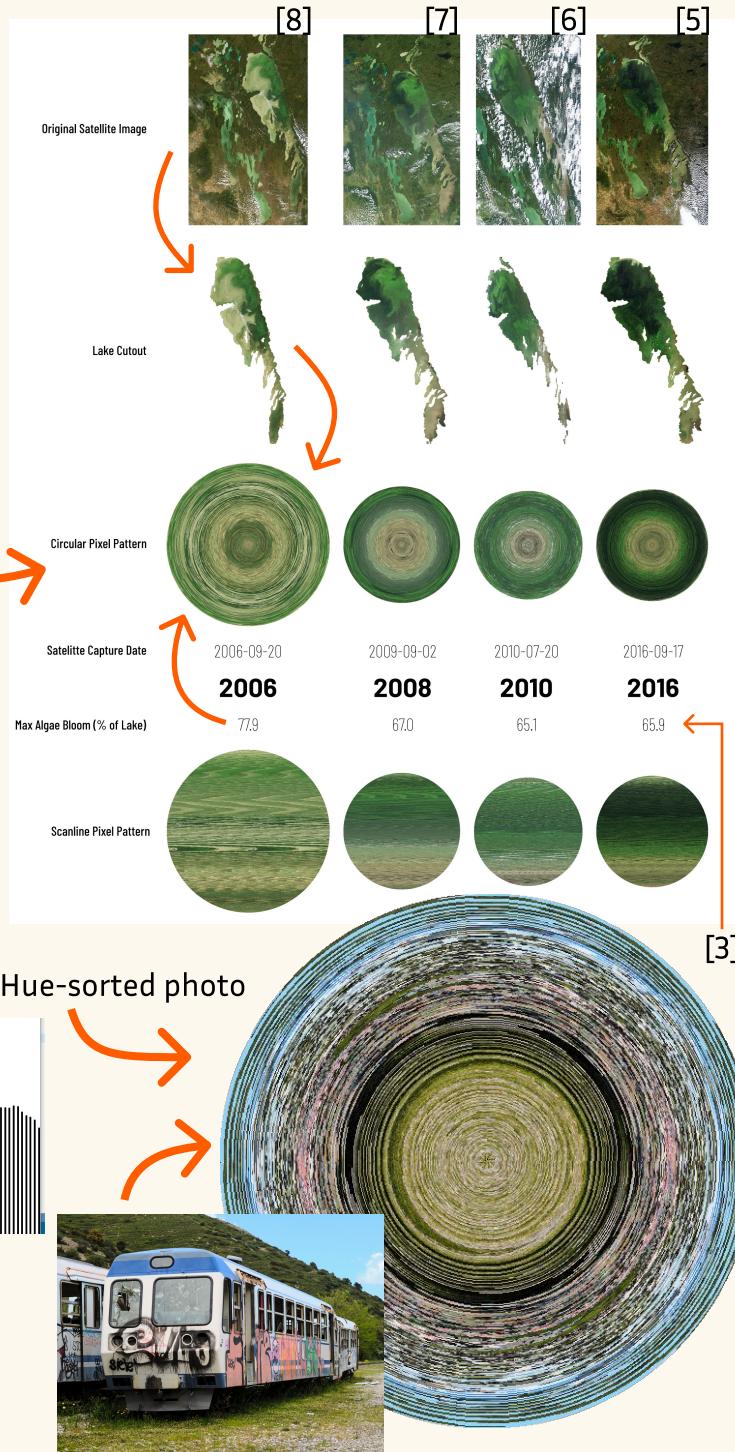
Shape & colour inversion exploration



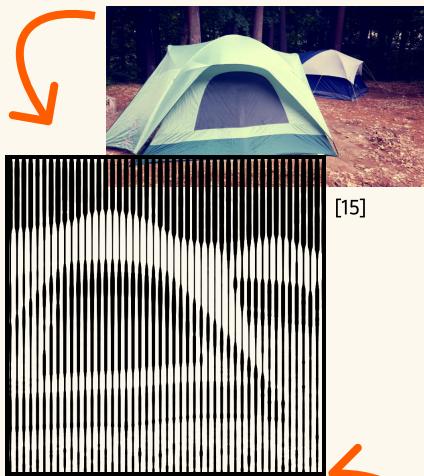
Wave code with noise

```
void Wave() {
    for (int i = 0; i<(pointsAmt*mod); i++) {
        float noiseVal = noise((mod+i)*noiseScale, mod*noiseScale);
        float x = i*pointsWdth;
        float y = sin(pointsAngle * i) * waveHeight;
        line(x, (height/2)+y+noiseVal*80, x, height);
    }
}
```

[4, 12]

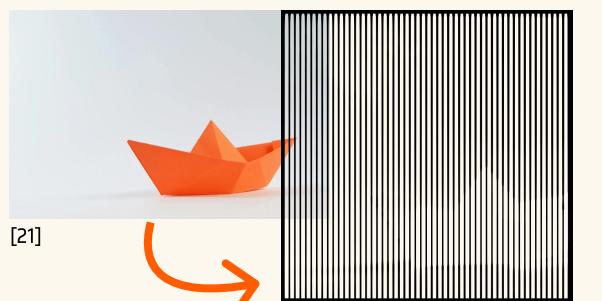


# CODE ELABORATION



Map function to map brightness to line thickness value

Issue with brightness:  
Similar brightnesses get filtered out, even-though it visually looks different



Create an outline around the image to aid in vinyl cutting

## Finalised code

```
import processing.svg.*;

PImage img; //define PImage
float varSpace[] = {15, 20, 25, 17.5};
float varThick[] = {1.5, 2.1, 2.7};
float transWidth;
float transHeight;

void setup() {
    size(800, 800); //define screen size
    for (int varSp = 0; varSp < 1; varSp++) {
        for (int varTh = 0; varTh < 1; varTh++) {
            for (int imgNr=2; imgNr <=2; imgNr++) {
                beginRecord(SVG, "Image "+str(imgNr)+", Sp"+str(varSpace[varSp])+", Th"+str(varThick[varTh])+".svg");
                String imgStr = str(imgNr)+".png"; //images in the drive only use number id
                img = loadImage(imgStr);
                pushMatrix();
                transWidth = (800-img.width)/2;
                transHeight = (800-img.height)/2;
                translate(transWidth, transHeight);
                img.loadPixels();
                for (int i = 0; (i < (img.width)); i += varSpace[varSp]) { //make the following code cycle for every column of pixels
                    float StrokeInit = 8; //Base line thickness
                    float initBright = brightness(img.pixels[i]); //calls the first pixel of every row (initial Brightness)
                    for (int j = 0; j < (img.height); j++) {
                        float pixBright = brightness(img.pixels[(img.width)*i+j]); //calls the every pixel in the specified row (from the parent loop) (pixels are stored in array, nr in first row is pixel, nr in second row is (image width + pixel)
                        float mapBright = map(pixBright, 0, 255, 7*varThick[varTh], varThick[varTh]);
                        push();
                        //references [1,2]
                        StrokeInit = lerp(StrokeInit, mapBright, 0.1);
                        strokeWeight(StrokeInit);
                        if (StrokeInit > 0) {
                            line(i+5, j, i+5, j+1);
                        }
                        pop();
                        initBright = pixBright;
                    }
                    popMatrix();
                    Border();
                    endRecord();
                }
            }
        }
    }
}

push();
//references [1,2]
StrokeInit = lerp(StrokeInit, mapBright, 0.1);

strokeWeight(StrokeInit);
if (StrokeInit > 0) {
    line(i+5, j, i+5, j+1);
}
pop();
initBright = pixBright;
}
popMatrix();
Border();
endRecord();
}

push();
//references [1,2]
StrokeInit = lerp(StrokeInit, mapBright, 0.1);

strokeWeight(StrokeInit);
if (StrokeInit > 0) {
    line(i+5, j, i+5, j+1);
}
pop();
initBright = pixBright;
}
popMatrix();
Border();
endRecord();
}

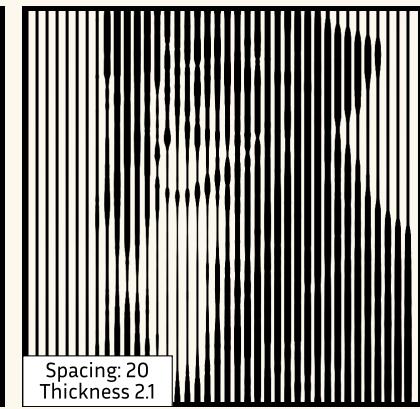
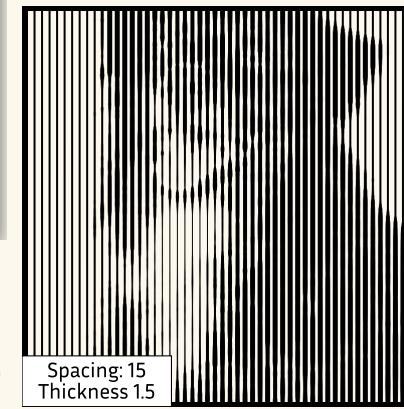
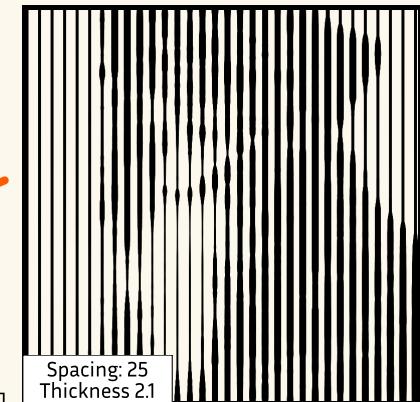
push();
//references [1,2]
StrokeInit = lerp(StrokeInit, mapBright, 0.1);

strokeWeight(StrokeInit);
if (StrokeInit > 0) {
    line(i+5, j, i+5, j+1);
}
pop();
initBright = pixBright;
}
popMatrix();
Border();
endRecord();
}
```

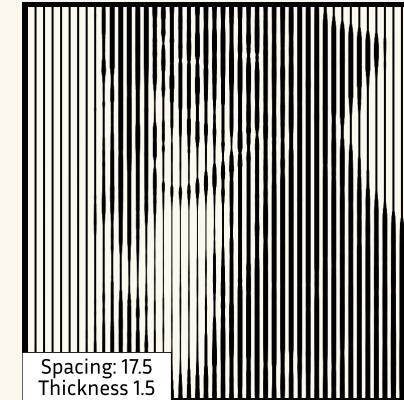
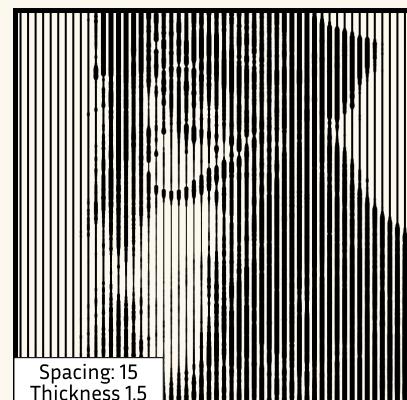
## Exploring variables:

- Line spacing
- Line thickness
- Number of lines

Interesting variations in visualisations allow us to balance between visual clarity and physical feasibility



Lerp function for scaling line thickness with the previous thickness (smoother lines)



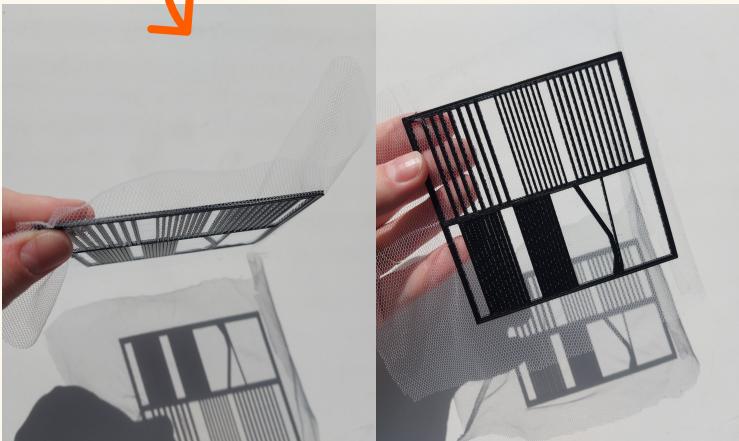
```
void Border() {
    strokeWeight(10);
    line(5, 0, 5, height);
    line(0, 5, width, 5);
    line(width-5, 0, width-5, height);
    line(width, height-5, 0, height-5);
}
```

## REALIZATION EXPLORATIONS

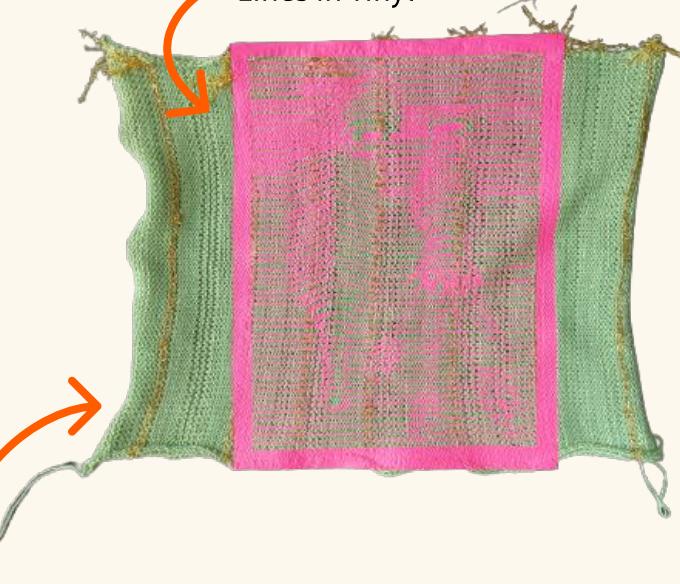
Realization experimentation shaped our design in certain manners, those will be explained here.

The corners of the blue spaces are places where the memories were made resulting in a pattern that can be repeated through out the whole bag

Laser cutting lines:  
too fragile to share the memories

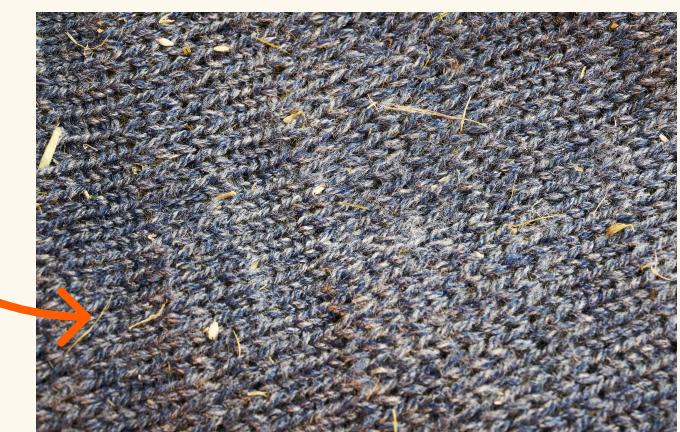


Lines in vinyl



The loop lengths of the knits is determined by the coordinates of the place the memory is made

A variety of connecting ways were explored for the memories. The button was the most appealing as they allow freedom in adjustment



## SHAPE EXPLORATIONS



3 compartments  
that can unfold



Revealing and  
hiding the body



Transformable circles

3 compartments,  
not foldable anymore,  
working with circles



Revealing and  
hiding the body



Circled shape with  
memory layers  
that can *revealed*  
or hidden



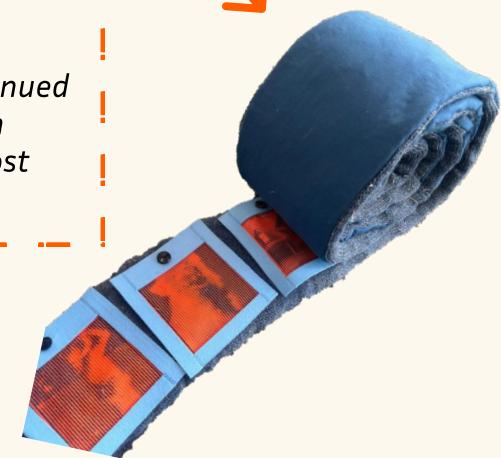
Through exploration  
it was revealed that the  
inner circle was most  
reachable, therefore...



Rolled up circle and when  
unfolded, *revealing* all its  
memory layers



All of our explorations are big in size. In the beginning, this was the result of 'being prepared for everything! This method of working seamlessly continued when working with memories, which eventually created the space the most dearest memories deserve.



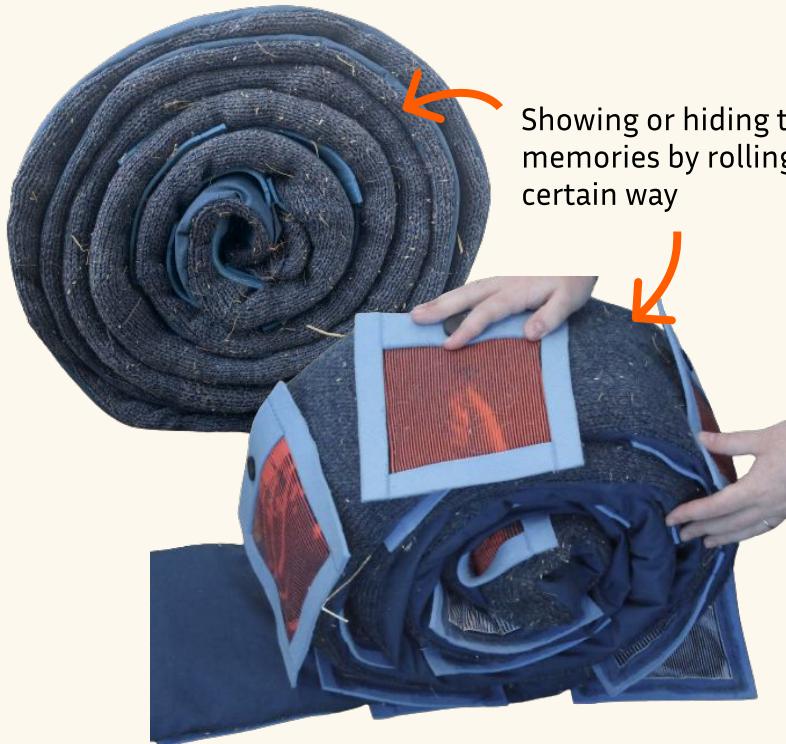
# FINAL DESIGN

## Creating the bag can be done in multiple ways:

- If you know the person will come to die soon, you can create it together as a goodbye ritual.
- If the person has already passed, the bag can be made individually or together in the first stage of grief.
- The making can function as a meditation ritual, especially the knitting.

## Using the bag

The heaviness of the bag in the beginning gives way for sharing the grief by caring it together. Later on, the bag can be managed on your own but shared through the memories of the lost one.



## Organizing the memories



Carrying it around is heavy and limits you in your abilities



Different ways of carrying



The interchangeable parts also allow you to share the images with other without having to show all the memories

## REFERENCES

- [1] amzon-ex. 2020. Draw a line of varying width in p5.js. Stack Overflow. Retrieved May 17, 2024 from <https://stackoverflow.com/q/62600387>
- [2] Casey Reas & Ben Fry. 2017. Processing 2.x and 3.x Forum. Processing 2.0 Forum. Retrieved May 17, 2024 from <https://forum.processing.org/two/discussion/22868/how-to-make-an-increasingly-thicker-line>
- [3] Environment and Climate Change Canada. Lake Winnipeg Satellite-derived Daily Algal Bloom Indices - Open Government Portal. Retrieved June 26, 2024 from <https://open.canada.ca/data/en/dataset/8cf1f5e8-021b-4000-a9b6-66a6339e8c2b>
- [4] George Profenza. 2016. Answer to "animating sine waves in processing." Stack Overflow. Retrieved June 13, 2024 from <https://stackoverflow.com/a/40295881>
- [5] Satellite Images 2016 – Lake Winnipeg Research Consortium. Retrieved June 26, 2024 from <https://www.lakewinnipegresearch.org/satellite-images-2016/>
- [6] Satellite Images 2010 – Lake Winnipeg Research Consortium. Retrieved June 26, 2024 from <https://www.lakewinnipegresearch.org/satellite-images-2010/>
- [7] Satellite Images 2009 – Lake Winnipeg Research Consortium. Retrieved June 26, 2024 from <https://www.lakewinnipegresearch.org/satellite-images-2009/>

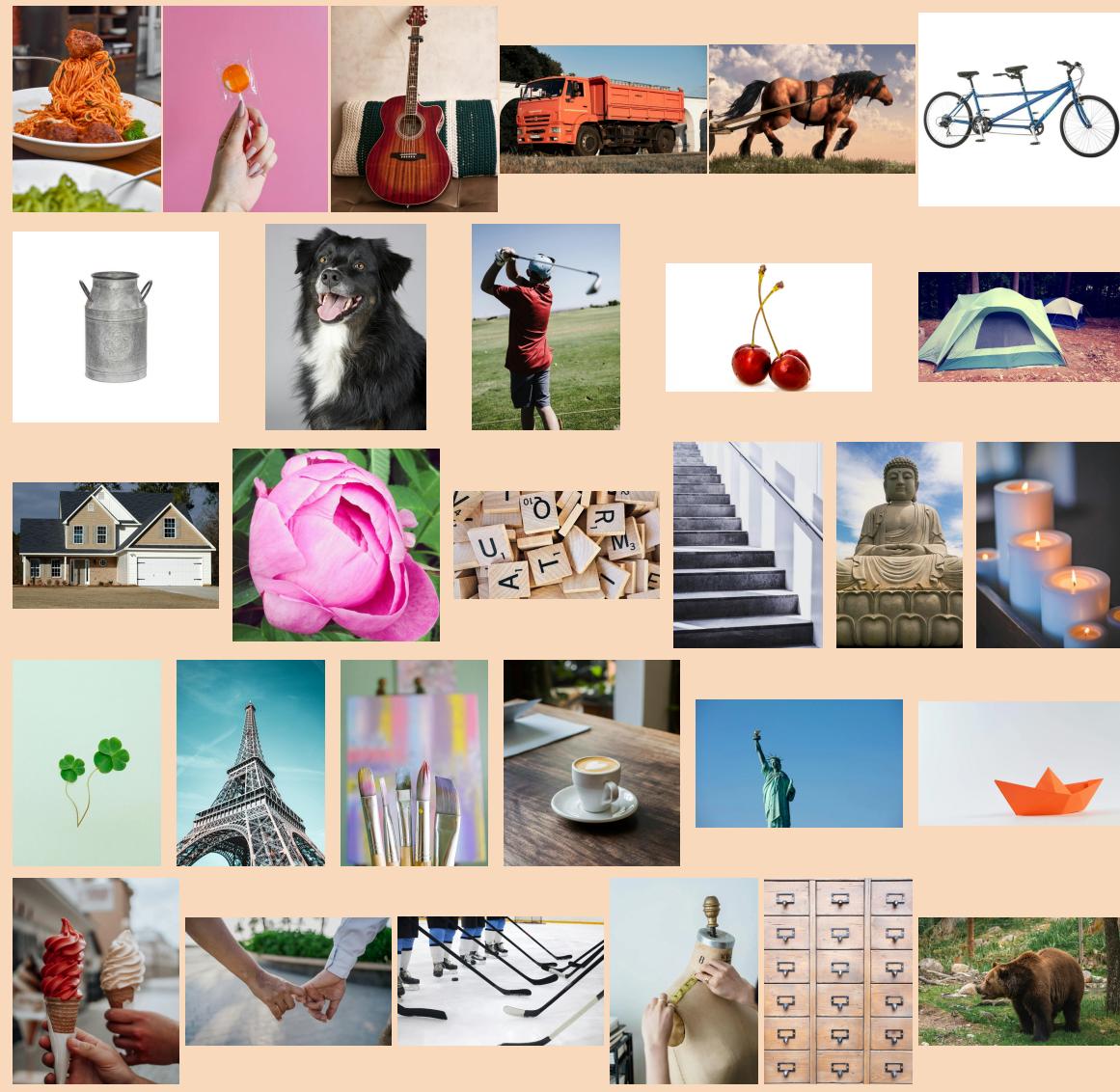
## BAG IMAGE REFERENCES

- [8] Satellite Images 2006 – Lake Winnipeg Research Consortium. Retrieved June 26, 2024 from <https://www.lakewinnipegresearch.org/satellite-images-2006/>
- [9] Zonsondergang Palmboom Laguna - Gratis foto op Pixabay. Retrieved June 13, 2024 from <https://pixabay.com/nl/photos/zonsondergang-palmboom-laguna-2164985/>
- [10] Foto door Tim Gouw op Pexels. Pexels. Retrieved June 13, 2024 from <https://www.pexels.com/nl-nl/foto/wit-en-veelkleurige-veerboten-2280358/>
- [11] Photo by Mihman Duğanlı on Pexels. Pexels. Retrieved June 13, 2024 from <https://www.pexels.com/photo/pods-of-ferris-wheel-19559185/>
- [12] noise() / Reference. Processing. Retrieved June 13, 2024 from [https://processing.org/reference/noise\\_.html](https://processing.org/reference/noise_.html)
- [13] 2019. A muscular horse with a chestnut coat and dark mane and tail pulls a... iStock. Retrieved June 26, 2024 from <https://www.istockphoto.com/nl/foto/werk-paard-gm1130704249-299131503>
- [14] Cosy @ Home Melkkan Embossed Logo Grijs 22,5x21xh35c - 5400586399706 4421265 BestSale. Retrieved June 26, 2024 from <https://www.bestsale.be/nl/cosy-home-melkkan-embossed-logo-grijs-22-5x21xh35c-5400586399706-4421265.html>
- [15] Foto door Raj Tatavarthy op Pexels. Pexels. Retrieved June 26, 2024 from <https://www.pexels.com/nl-nl/foto/witgroene-en-zwarте-buitentent-111362/>
- [16] Foto door Hannah Gibbs op Pexels. Pexels. Retrieved June 26, 2024 from <https://www.pexels.com/nl-nl/foto/vrijheidsbeeld-3615644/>
- [17] Foto door Tatjana op Pexels. Pexels. Retrieved June 26, 2024 from <https://www.pexels.com/nl-nl/foto/dier-beest-staand-buiten-7944483/>
- [18] Foto door Louis op Pexels. Pexels. Retrieved June 26, 2024 from <https://www.pexels.com/nl-nl/foto/lage-hoekfoto-van-de-eiffeltoren-1530259/>
- [19] Foto door Lisa Fotios op Pexels. Pexels. Retrieved June 26, 2024 from <https://www.pexels.com/nl-nl/foto/rood-kersenfruit-109274/>

## BAG IMAGE REFERENCES (CONT.)

- [20] Foto door Daniela Ruiz op Pexels. Pexels. Retrieved June 26, 2024 from <https://www.pexels.com/nl-nl/foto/gautama-boeddha-1042206/>
- [21] Foto door Miguel Á. Padriñán op Pexels. Pexels. Retrieved June 26, 2024 from <https://www.pexels.com/nl-nl/foto/oranje-papieren-boot-op-witte-ondergrond-19678/>
- [22] Foto door Tony Schnagl op Pexels. Pexels. Retrieved June 26, 2024 from <https://www.pexels.com/nl-nl/foto/spelers-players-atleten-sporters-6468730/>
- [23] Foto door Michael Burrows op Pexels. Pexels. Retrieved June 26, 2024 from <https://www.pexels.com/nl-nl/foto/iemand-persoon-creatief-kunstzinnig-7147968/>
- [24] Foto door Monstera Production op Pexels. Pexels. Retrieved June 26, 2024 from <https://www.pexels.com/nl-nl/foto/bruin-houten-kast-met-lade-7794448/>
- [25] Foto door Álvaro Llanos op Pexels. Pexels. Retrieved June 26, 2024 from <https://www.pexels.com/nl-nl/foto/muziekinstrument-akoestische-gitaar-snaarinstrument-verticaal-schot-13174694/>
- [26] Foto door Valeria Boltneva op Pexels. Pexels. Retrieved June 26, 2024 from <https://www.pexels.com/nl-nl/foto/selectieve-focusfotografie-van-kaarsen-1123256/>
- [27] Foto door Sergeich 03 op Pexels. Pexels. Retrieved June 26, 2024 from <https://www.pexels.com/nl-nl/foto/man-vent-kerel-voertuig-9280464/>
- [28] Foto door Anna Tarazevich op Pexels. Pexels. Retrieved June 26, 2024 from <https://www.pexels.com/nl-nl/foto/man-en-vrouw-hand-in-hand-4839774/>
- [29] Foto door Pixabay op Pexels. Pexels. Retrieved June 26, 2024 from <https://www.pexels.com/nl-nl/foto/stapel-scrabble-letterstukken-278888/>
- [30] Foto door Cree Payton op Pexels. Pexels. Retrieved June 26, 2024 from <https://www.pexels.com/nl-nl/foto/voedsel-eten-bord-plaat-9617397/>
- [31] Foto door Carlie Wright op Pexels. Pexels. Retrieved June 26, 2024 from <https://www.pexels.com/nl-nl/foto/voedsel-eten-hand-met-de-hand-10562800/>
- [32] Foto door JorgeArturo Andrade op Pexels. Pexels. Retrieved June 26, 2024 from <https://www.pexels.com/nl-nl/foto/trappen-grijstinten-fotografie-921025/>
- [33] Foto door Deeana Arts op Pexels. Pexels. Retrieved June 26, 2024 from <https://www.pexels.com/nl-nl/foto/ondiepe-focus-foto-van-verfborstels-1646953/>
- [34] Foto door Anna Shvets op Pexels. Pexels. Retrieved June 26, 2024 from <https://www.pexels.com/nl-nl/foto/geluk-mazzel-fabriek-plant-3876633/>
- [35] Foto door Jopwell op Pexels. Pexels. Retrieved June 26, 2024 from <https://www.pexels.com/nl-nl/foto/man-swingende-golfclub-geconfrontereerd-met-grasveld-1325653/>
- [36] Foto door Lucie Liz op Pexels. Pexels. Retrieved June 26, 2024 from <https://www.pexels.com/nl-nl/foto/handen-toetje-vasthouden-ijsje-4698506/>
- [37] Foto door Pixabay op Pexels. Pexels. Retrieved June 26, 2024 from <https://www.pexels.com/nl-nl/foto/witte-en-zwarte-hond-met-lange-vacht-220938/>
- [38] Foto door Pixabay op Pexels. Pexels. Retrieved June 26, 2024 from <https://www.pexels.com/nl-nl/foto/bruin-en-wit-houten-huis-164558/>
- [39] Foto door Anna Tukhfatullina Food Photographer/Stylist op Pexels. Pexels. Retrieved June 26, 2024 from <https://www.pexels.com/nl-nl/foto/latte-in-beker-2648988/>
- [40] Pioenroos | Pioen. Pinterest. Retrieved June 26, 2024 from <https://co.pinterest.com/pin/pioenroos--380976449698928194/>
- [41] Pacific Baby Dualie Tandem Bicycle. Walmart.com. Retrieved June 26, 2024 from <https://www.walmart.com/ip/Pacific-Baby-Dualie-Tandem-Bicycle/150851430>

# APPENDIX 1: FINAL CODE & DATA



Bag IMAGE References [13–41]

```

import processing.svg.*;

PImage img; //define PImage
float imgx;
float imgy;
int linespace = 10;
float varSpace[] = {15, 20, 25, 17.5};
float varThick[] = {1.5, 2.1, 2.7};
float transwidth;
float transheight;

void setup() {
    //int screenW = displayWidth/3;
    //int screenH = displayHeight/3;
    size(800, 800); //define screen size
    for (int varSp = 0; varSp < 1; varSp++) {
        for (int varTh = 0; varTh < 1; varTh++) {
            for (int imgNr=0; imgNr <2; imgNr++) {

                beginRecord("Image "+str(imgNr)+", Sp="+str(varSpace[varSp])+", Th="+str(varThick[varTh])+".svg");

                String imgStr = "lineImage"+imgNr+".png";
                String imgStr2 = str(imgNr)+"_png"; //images in the drive only use number id
                img = loadImage(imgStr); //define the image to use (add _ or __ for variants)
                img.filter(TRESHOLD, 0.7); //testing with thresholds/greyscale

                //img.resize(screenW, screenH);
                //img.height = screenH;
                //img.resize(screenW,screenH);
                pushMatrix();

                transwidth = (800-img.width)/2;
                transheight = (800-img.height)/2;

                translate(transwidth, transheight);

                img.loadPixels();

                strokeCap(SQUARE);
                imgy = (height - img.height)/2;
                for (int i = 0; i < img.width(); i += varSpace[varSp]) { //make the following code cycle for every column of pixels
                    float StrokeInit = 8; //Base line thickness
                    //https://stackoverflow.com/questions/21005871/java-bufferedimage-get-single-pixel-brightness (tried, doesn't work)

                    float initBright = brightness(img.pixels[i]); //calls the first pixel of every row (initial Brightness)

                    for (int j = 0; j < (img.height); j++) {
                        //for (int i = 0; j > 0; j--) { //trying to calculate backwards, unsure how to)
                        float pixBright = brightness(img.pixels[(i*(img.width)+j)]);
                        //calls the every pixel in the specified row (from the parent loop
                        //pixels are stored in array, nr in first row is pixel, nr in second row is image width + pixel

                        //if (pixBright < 85){ //code to create a bit more of a gradient, literally increased generating time by like 5....
                            pixBright = 0;
                        }
                        else if (pixBright > 170){
                            pixBright = 255;
                        }
                        else {
                            pixBright = 127.5;
                        }

                        float mapBright = map(pixBright, 0, 200, 7*varThick[varTh], varThick[varTh]); //maps line thickness based on the pixel brightness
                        //brightness value 0 becomes line thickness 8, brightness 255 becomes thickness 0.3, the values in-between are MAPPED accordingly
                        //inverted colours are also pretty cool, though this does not display the imaging technique as well
                        //https://stackoverflow.com/questions/62600387/draw-a-line-of-varying-width-in-p5js
                        //https://forum.processing.org/two/discussion/22868/how-to-make-an-increasingly-thicker-line.html
                        push();
                        StrokeInit = lerp(StrokeInit, mapBright, 0.1); //use lerp to find line thickness for each pixel and compare this to previous thickness
                        strokeCap(SQUARE); //also creates interesting patterns, not very clean lines though

                        strokeWeight(StrokeInit);
                        if (StrokeInit > 0) {
                            line(+5, j, +5, j+1);
                        }
                        pop();

                        initBright = pixBright;
                        //println(img.pixels[i*img.width+i])
                        println(str(((i*img.width)+j+1)) + " of "+str(img.width*img.height)+", image "+str(imgNr)+",
                        +str(varSpace[varSp])+ "- "+str(varThick[varTh]));
                    }
                }
                popMatrix();
                Border();
                endRecord();
                print("done with image "+str(imgNr)+", Version: "+str(varSpace[varSp])+", "+str(varThick[varTh])+", Total versions: "+((varSp+1)*(varTh+1)));
            }
        }
    }
}

void draw() {
    // image(img, imgx, imgy); used initially to test PImage
}

void Border() {
    strokeWeight(10);
    line(5, 0, 5, height);
    line(0, 5, width, 5);
    line(width-5, 0, width-5, height);
    line(width, height-5, 0, height-5);
}

```

Finalised code  
(with commented code and all comments)