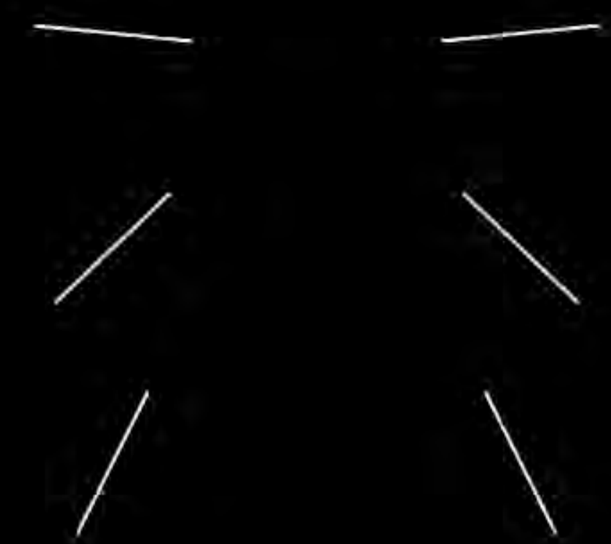
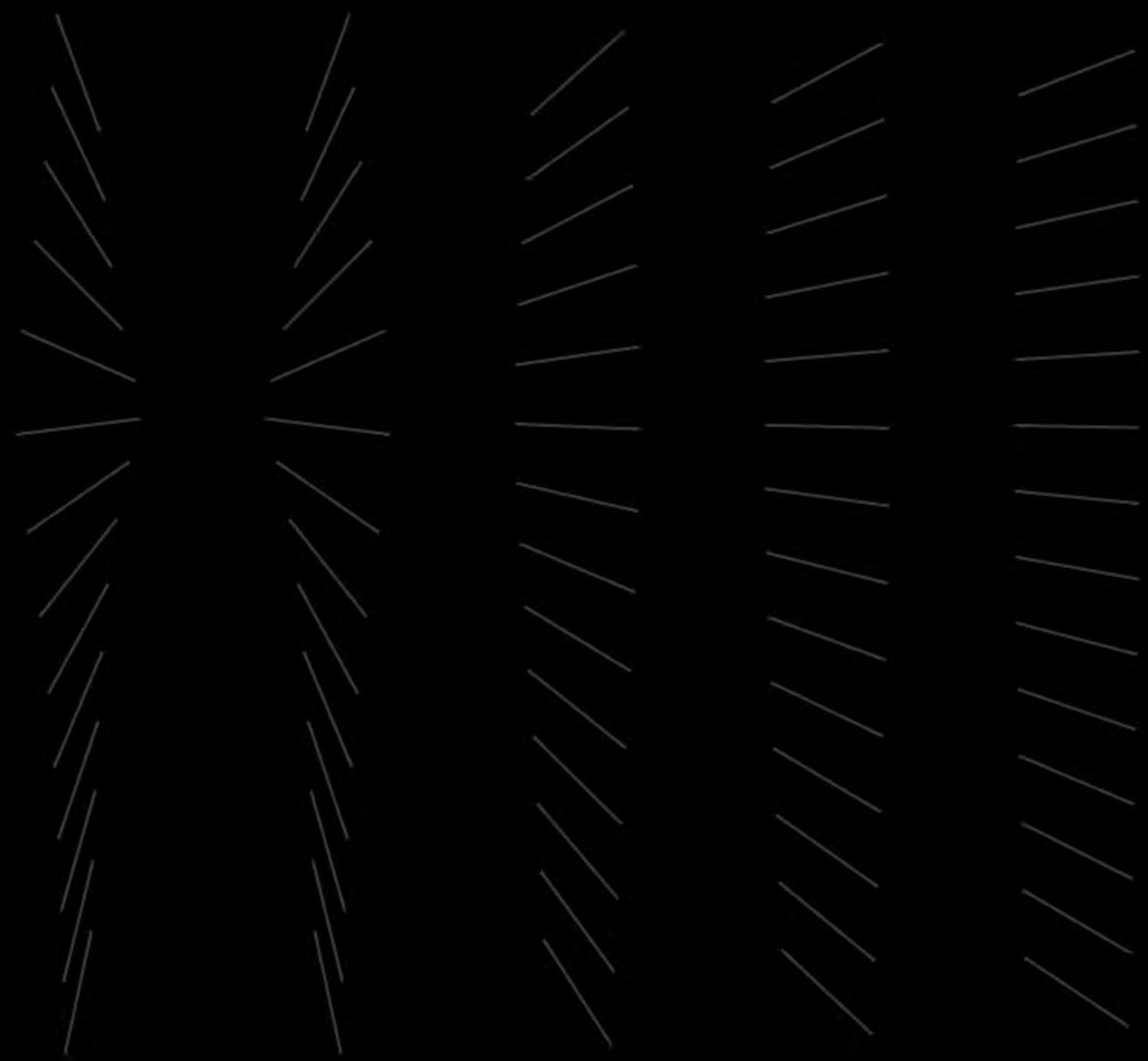
creative coding & generative art



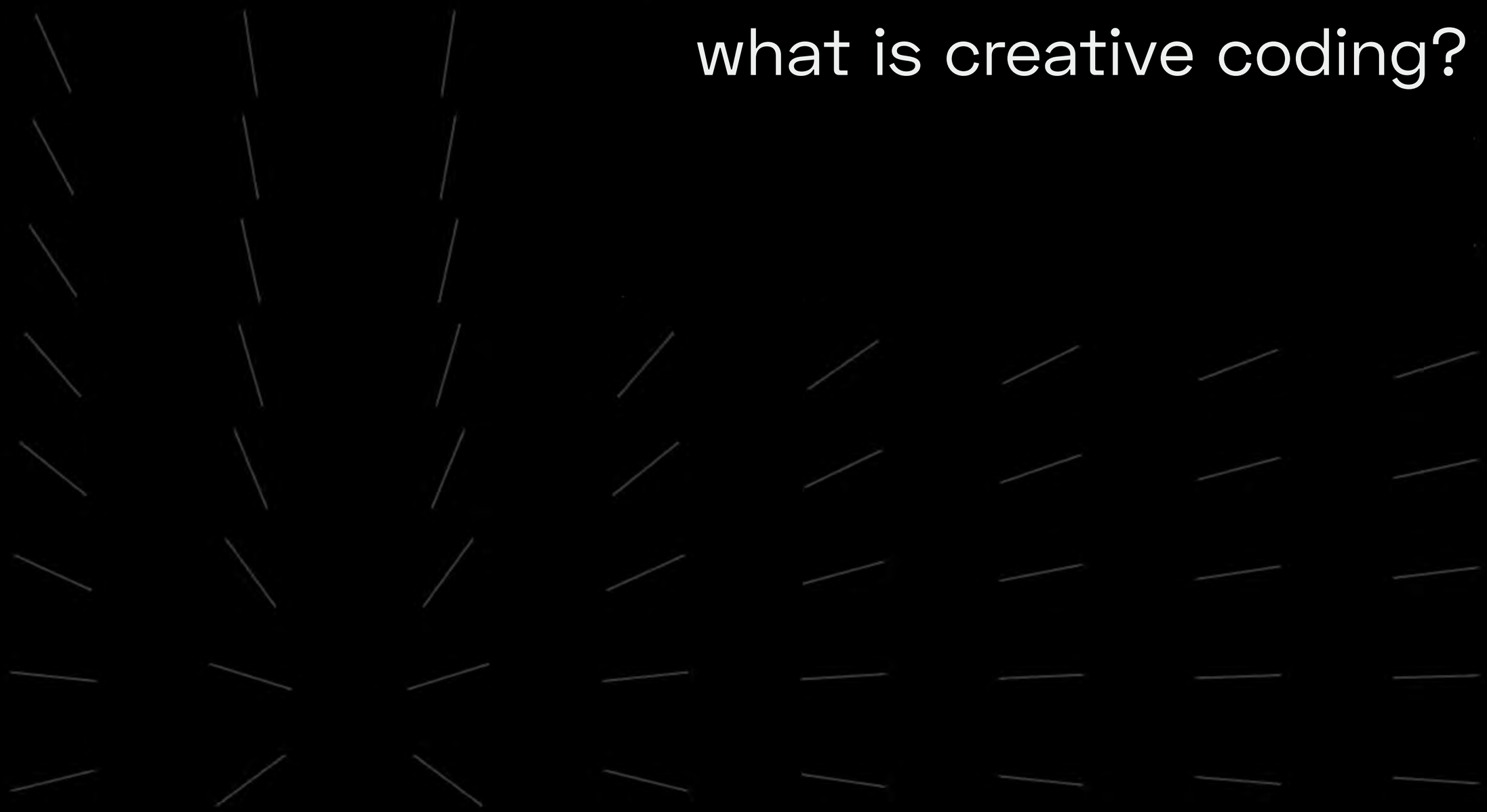
with Matt DesLauriers

about the workshop



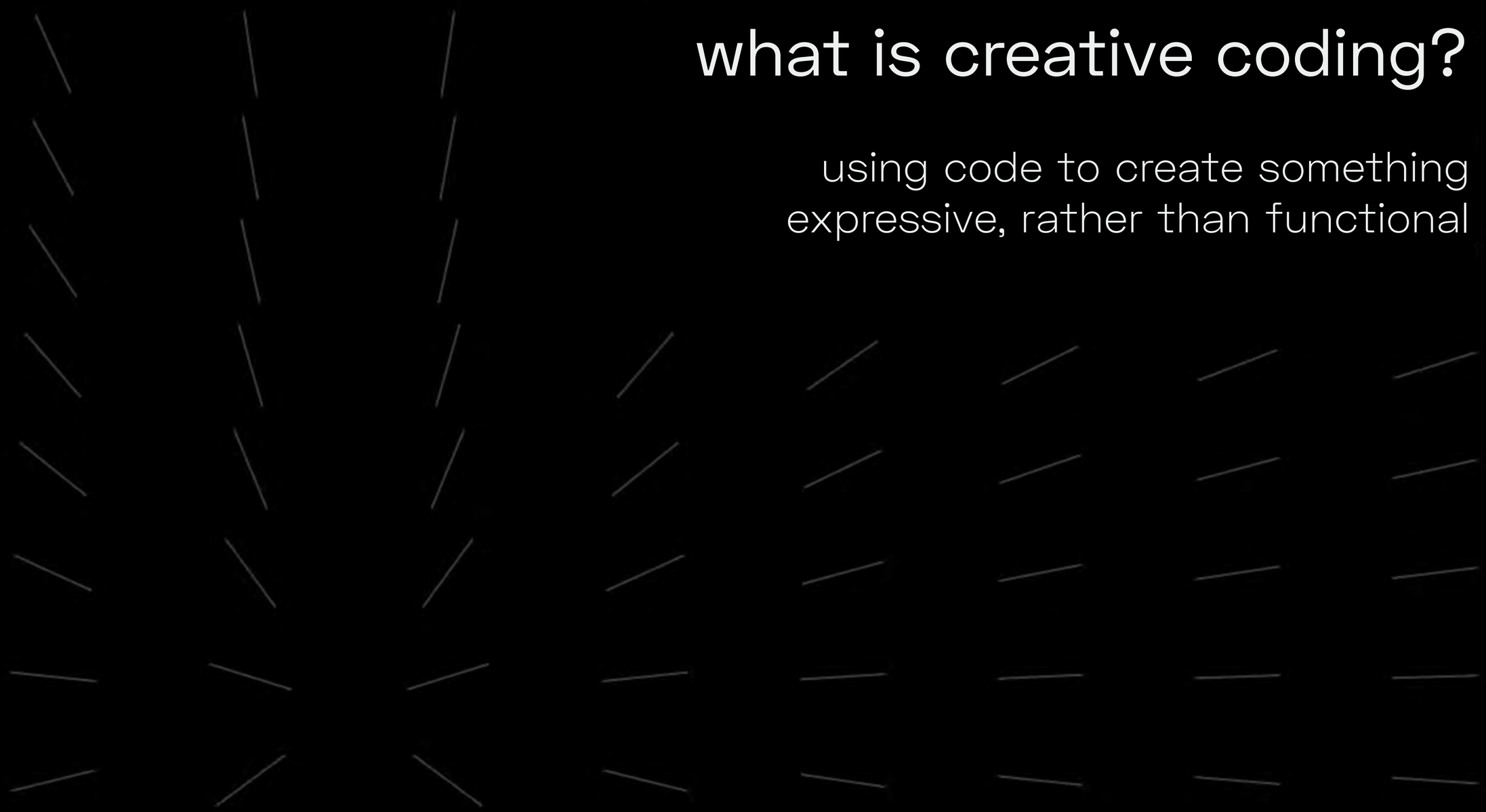
definitions

what is creative coding?

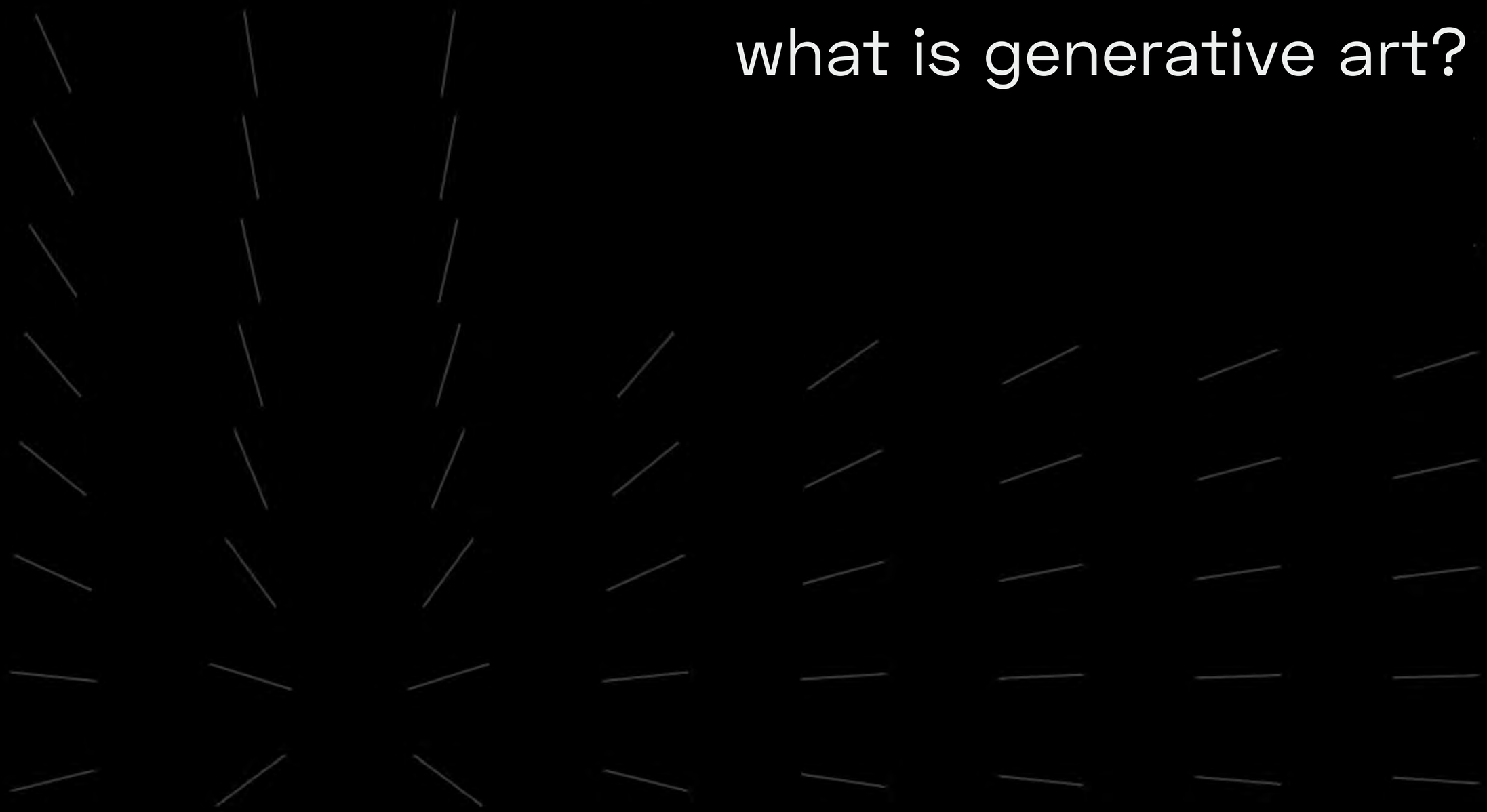


what is creative coding?

using code to create something
expressive, rather than functional



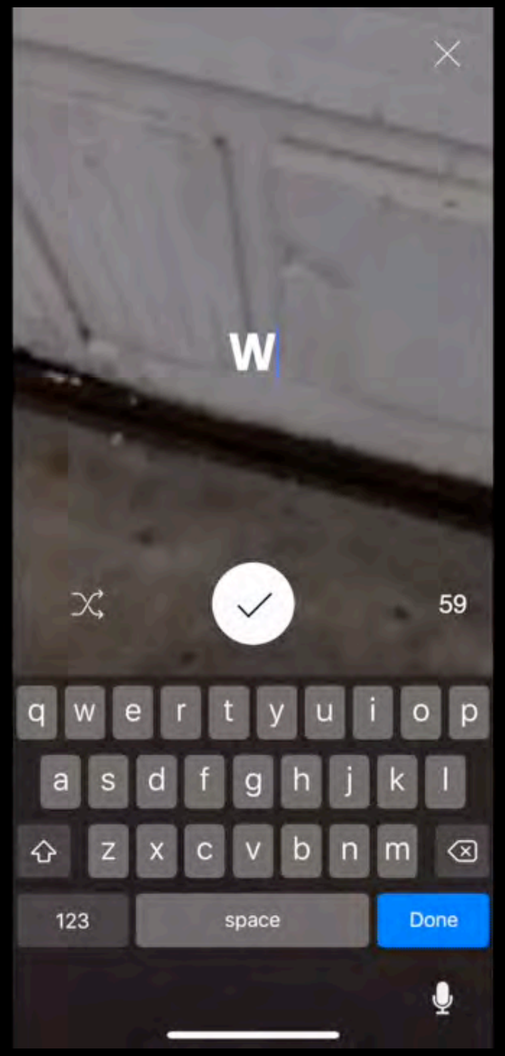
what is generative art?



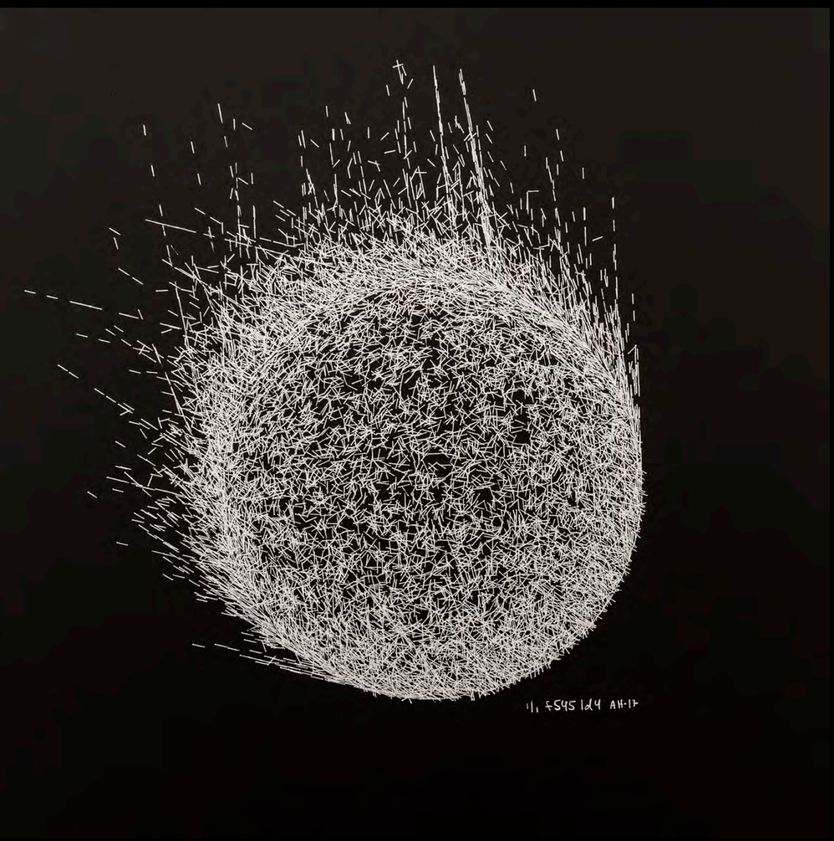
what is generative art?

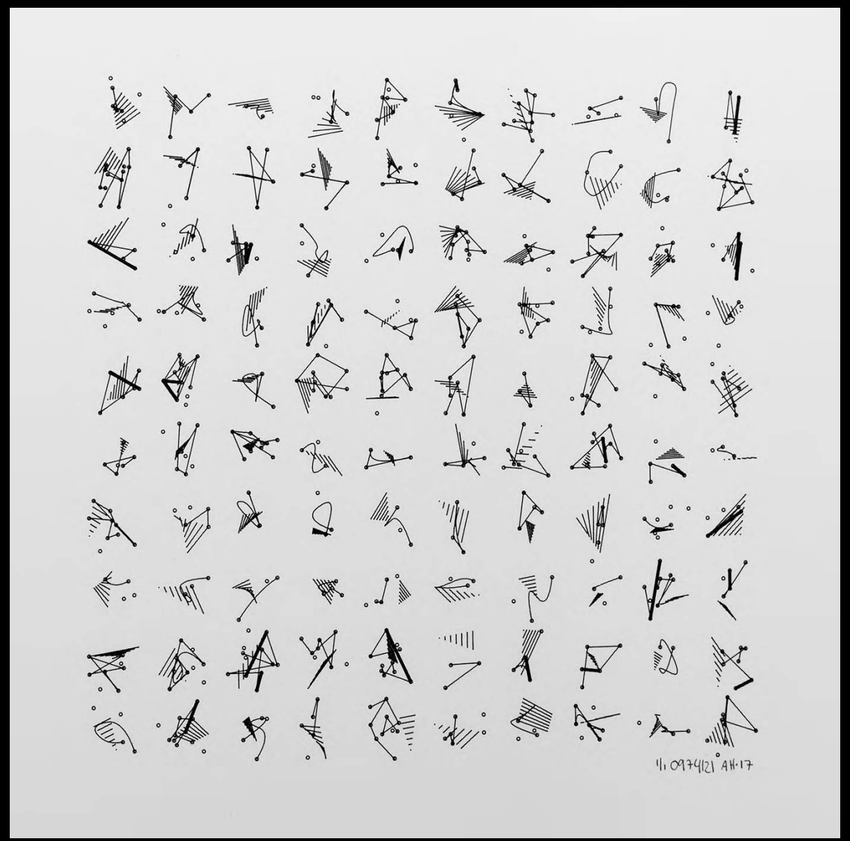
artworks created in part by an
autonomous or rule-based system

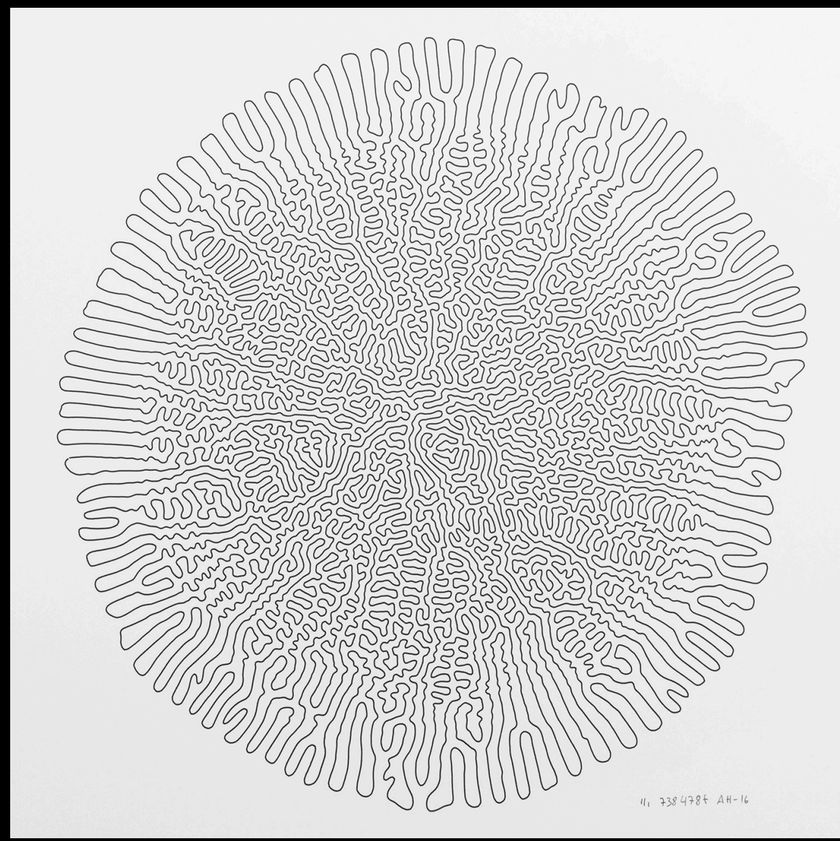
in the wild



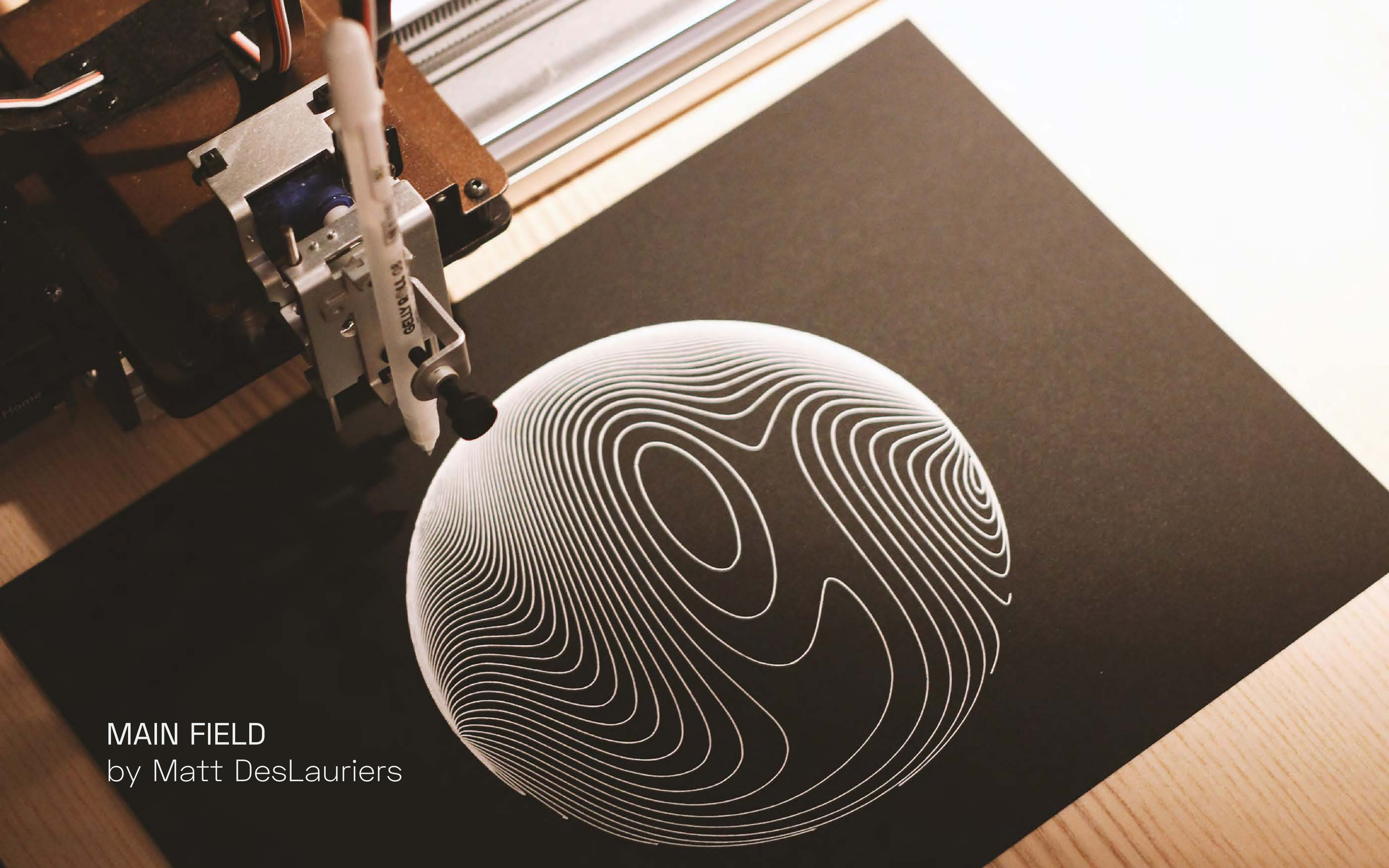
Weird Type
by Zach Lieberman



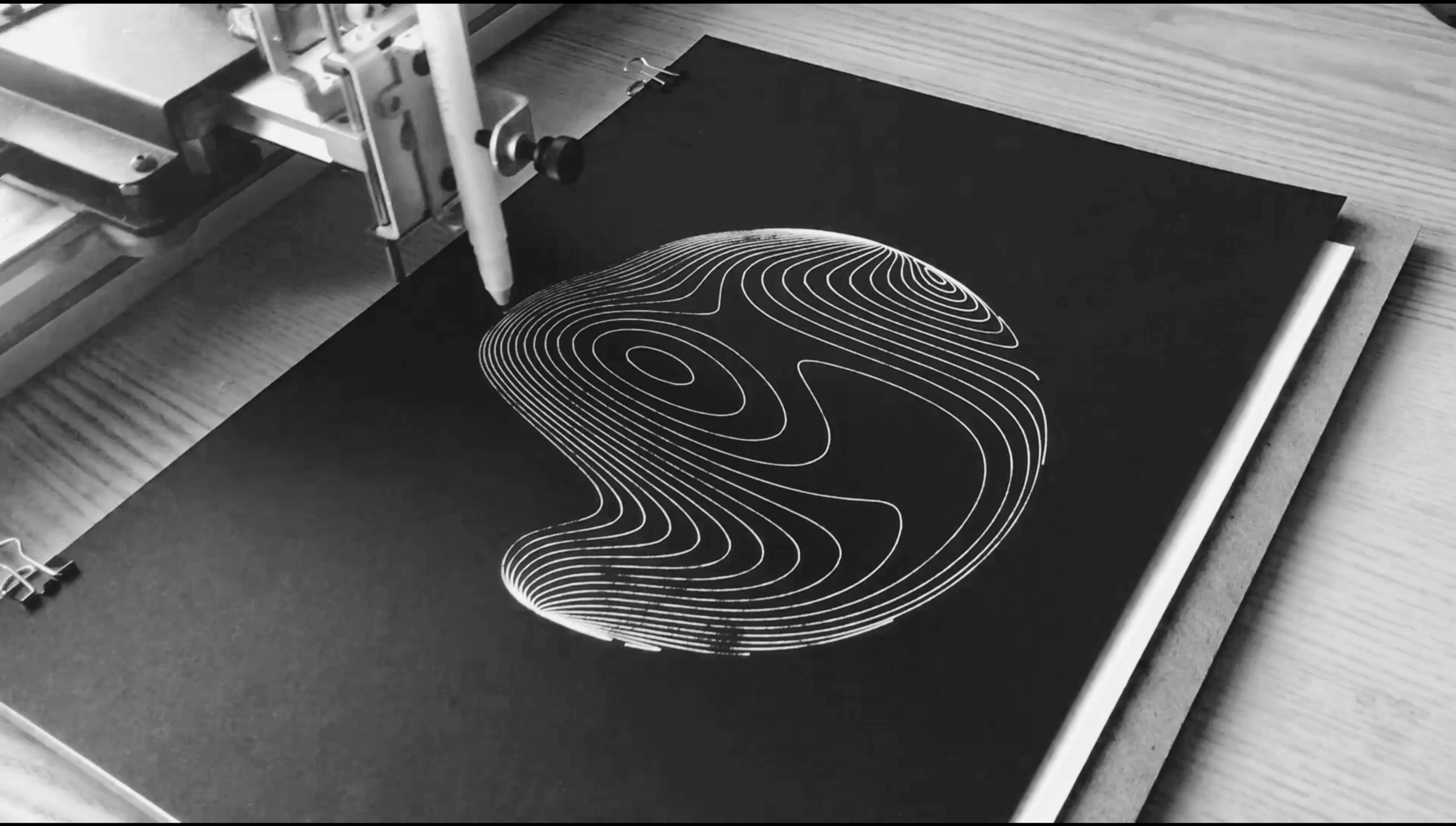




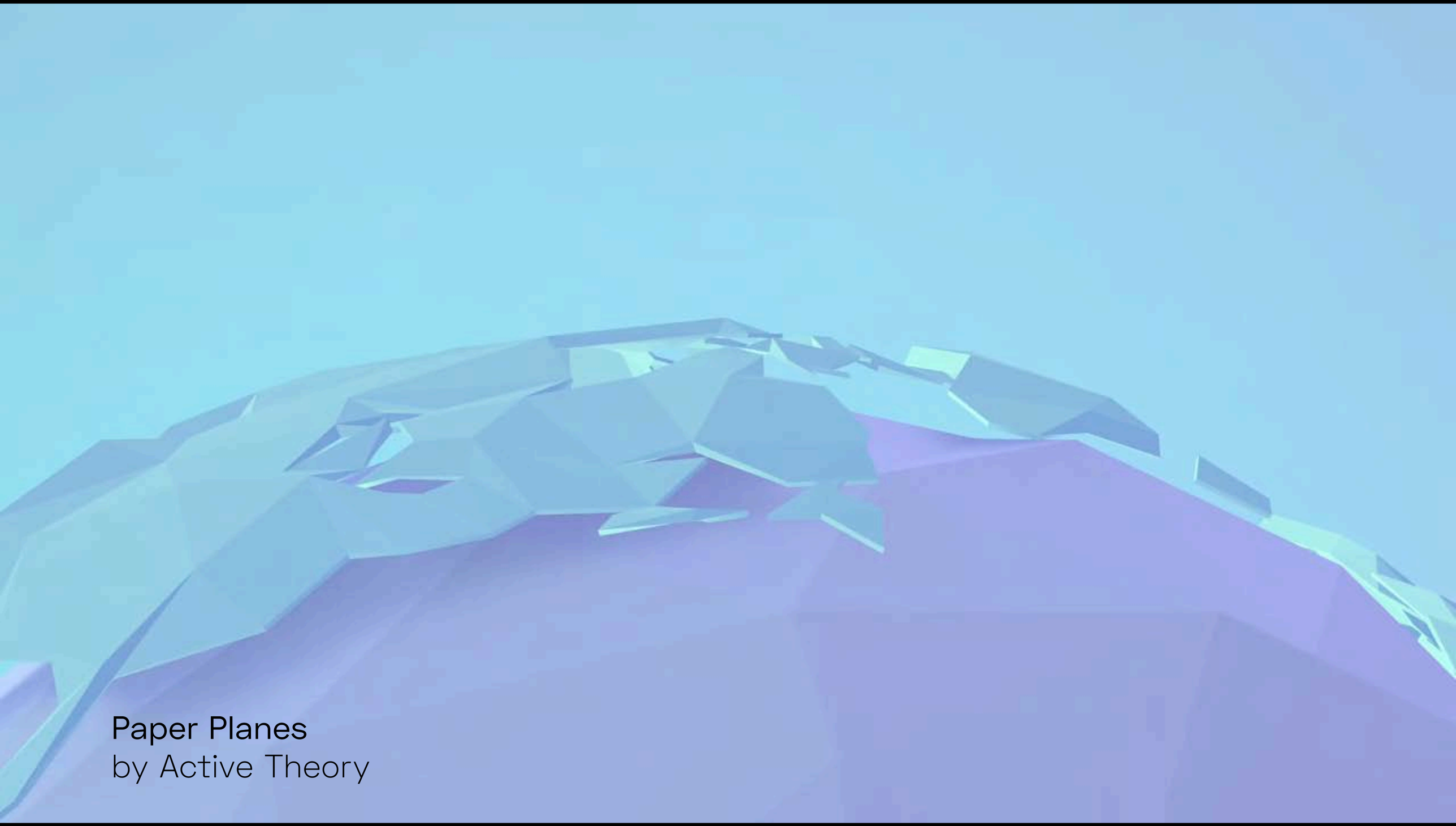
@inconvergent
Anders Hoff

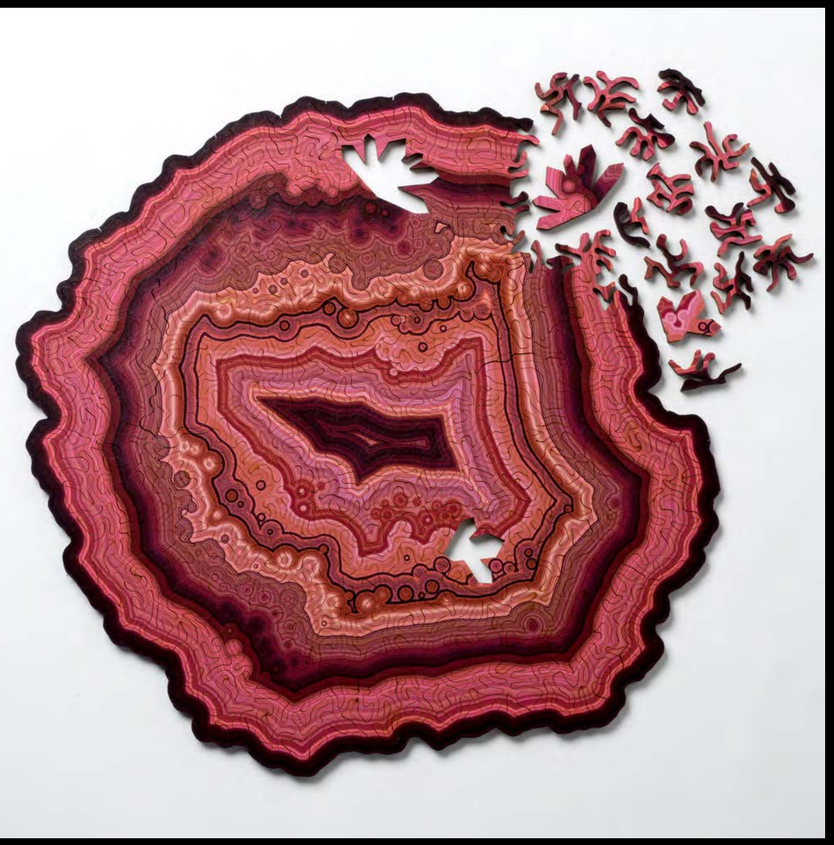


MAIN FIELD
by Matt DesLauriers



SUN
by Philip Schütte x Random Studio


Paper Planes
by Active Theory



Generative Puzzles, Housewares and other Products
Nervous System

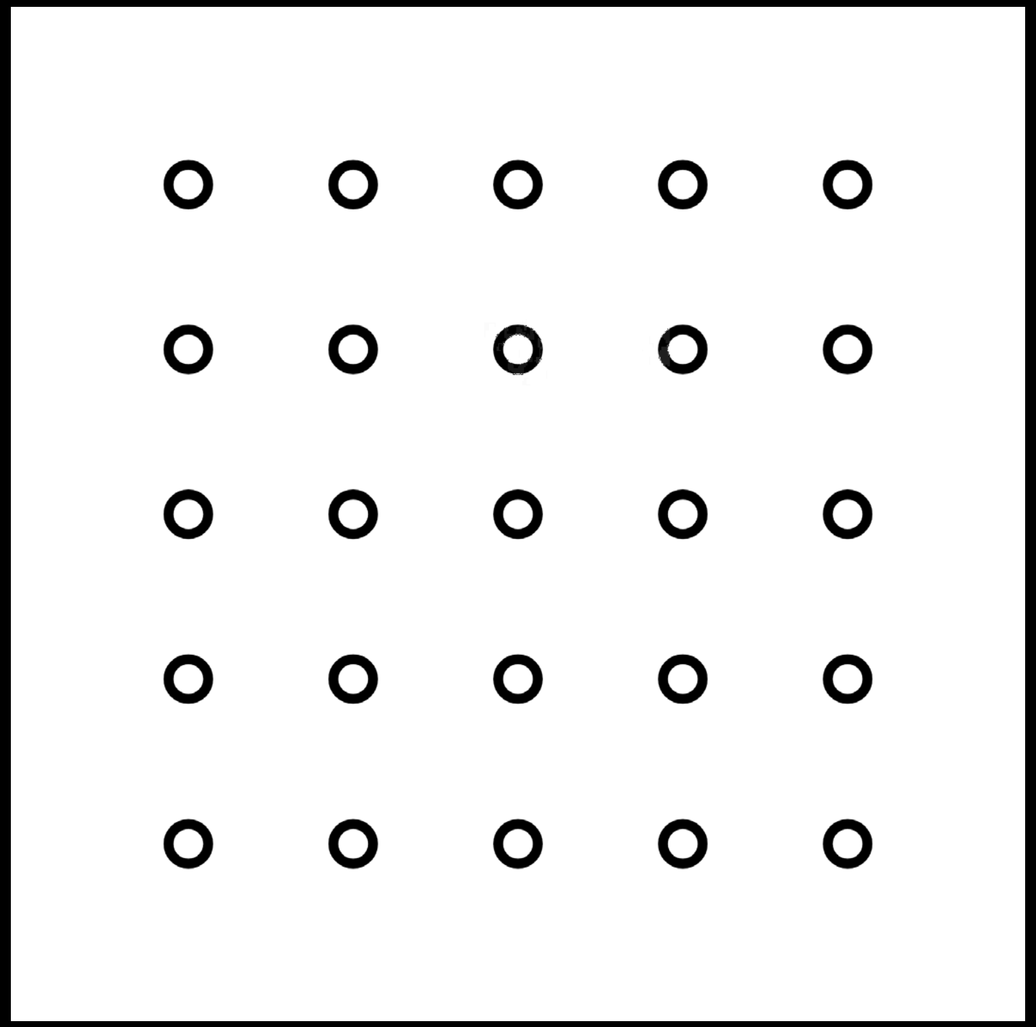
let’s make art!

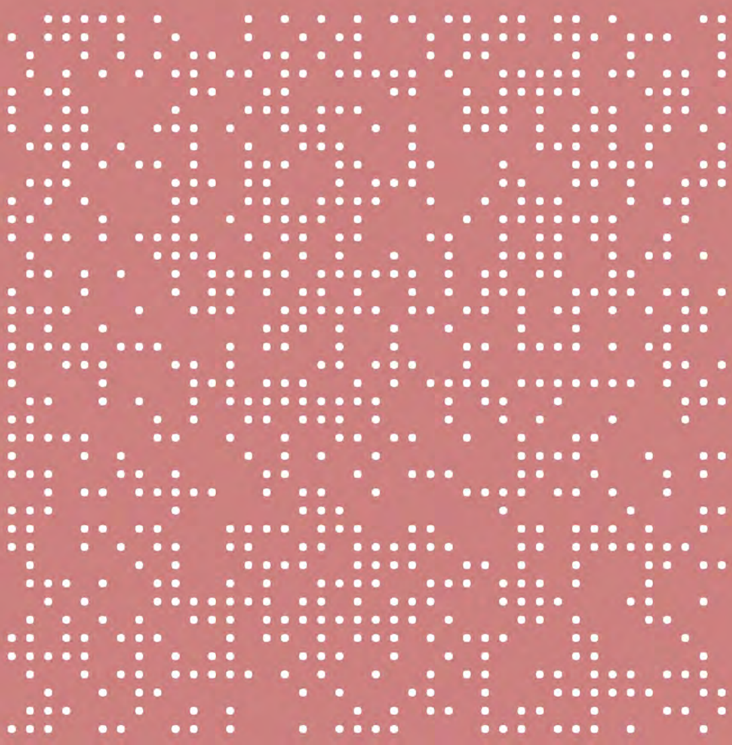
Unicode Compositions

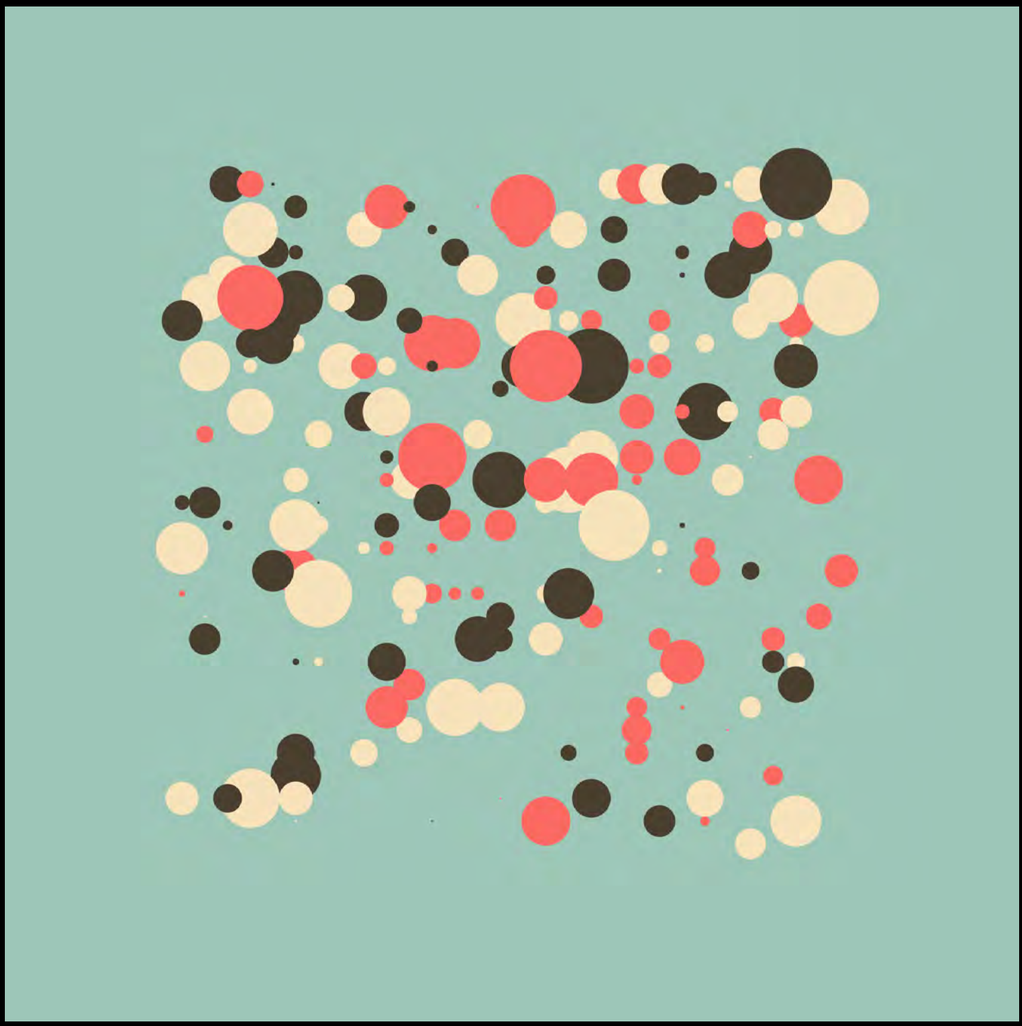
tools & setup

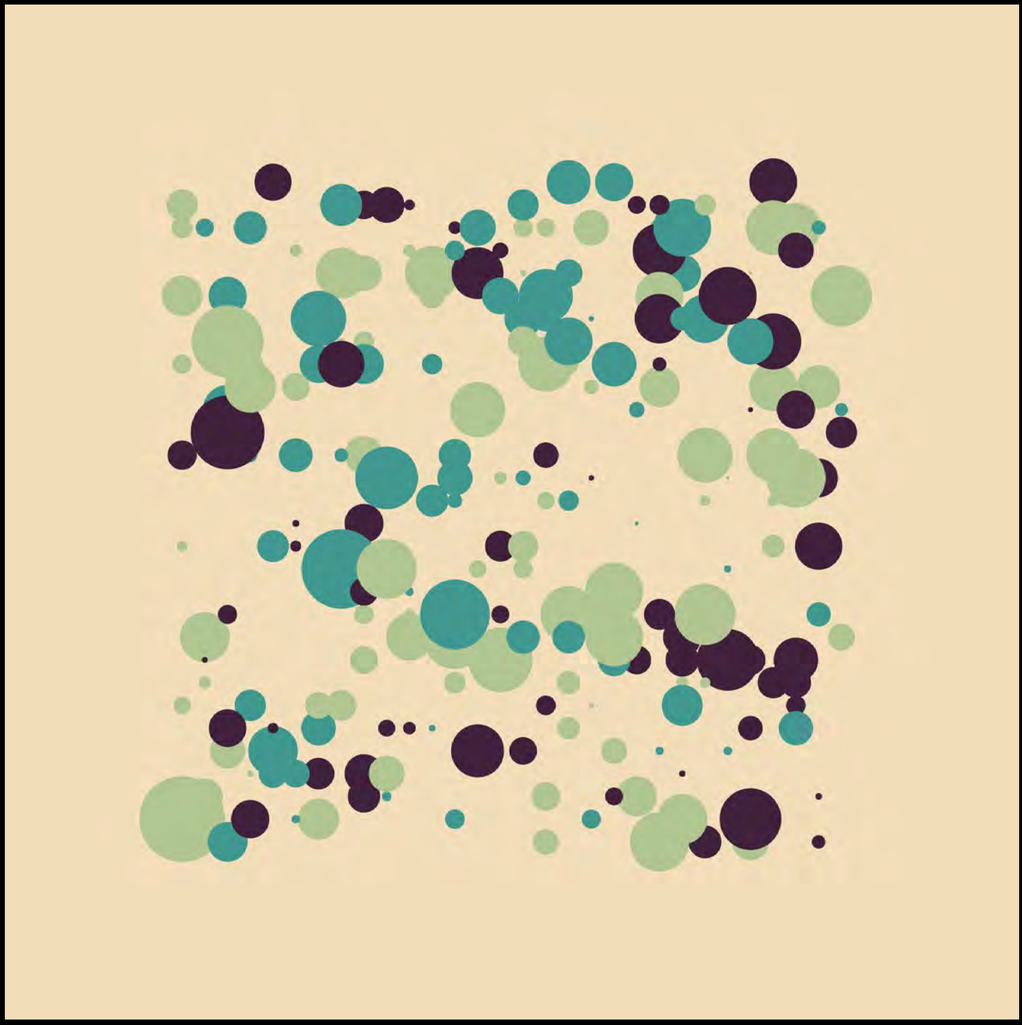
https://github.com/mattdesl/workshop-generative-art

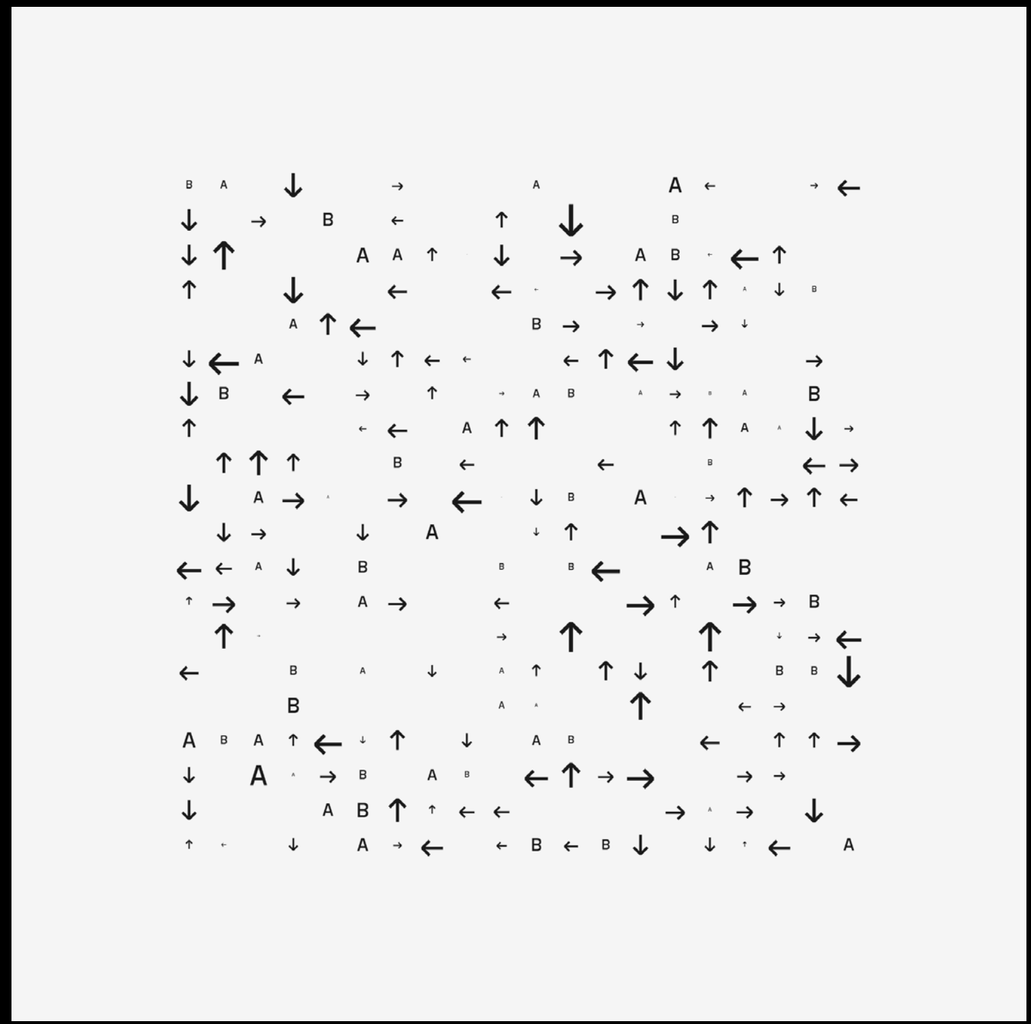
the grid

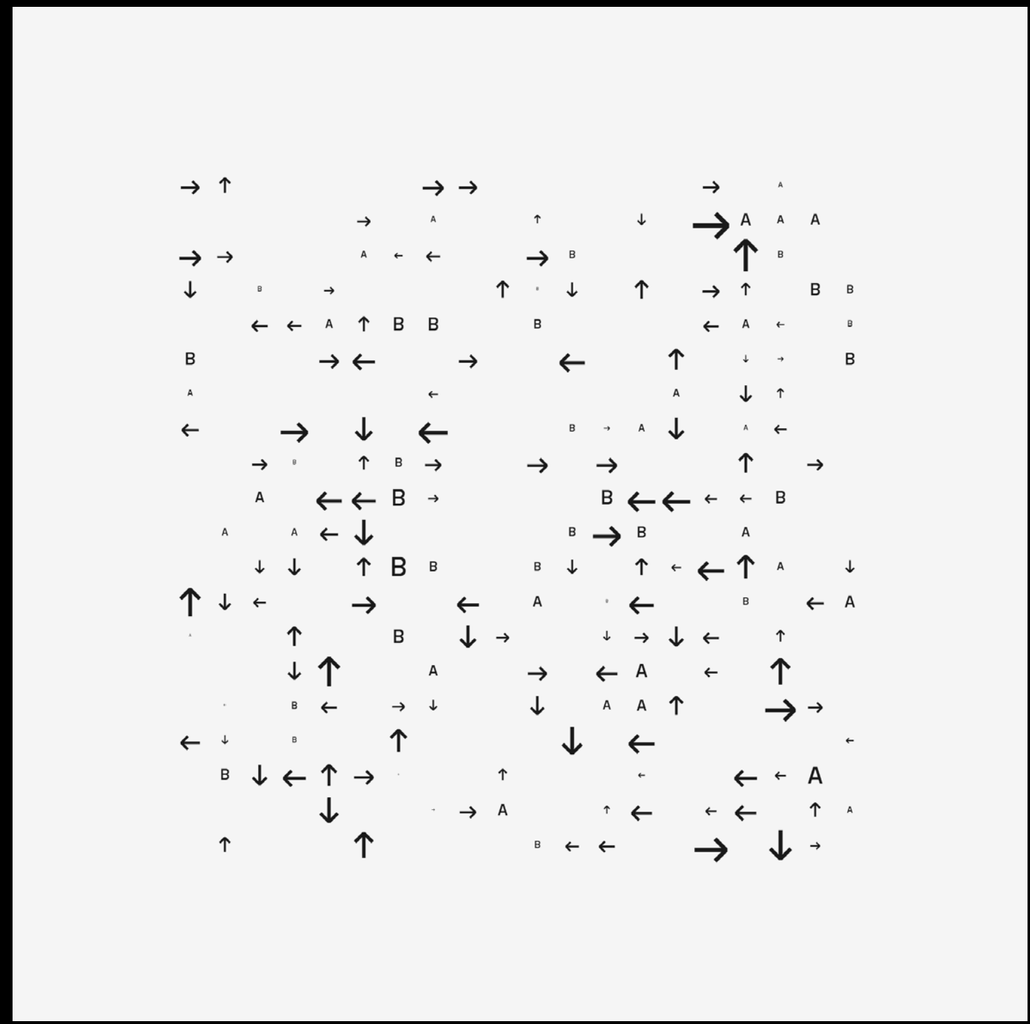








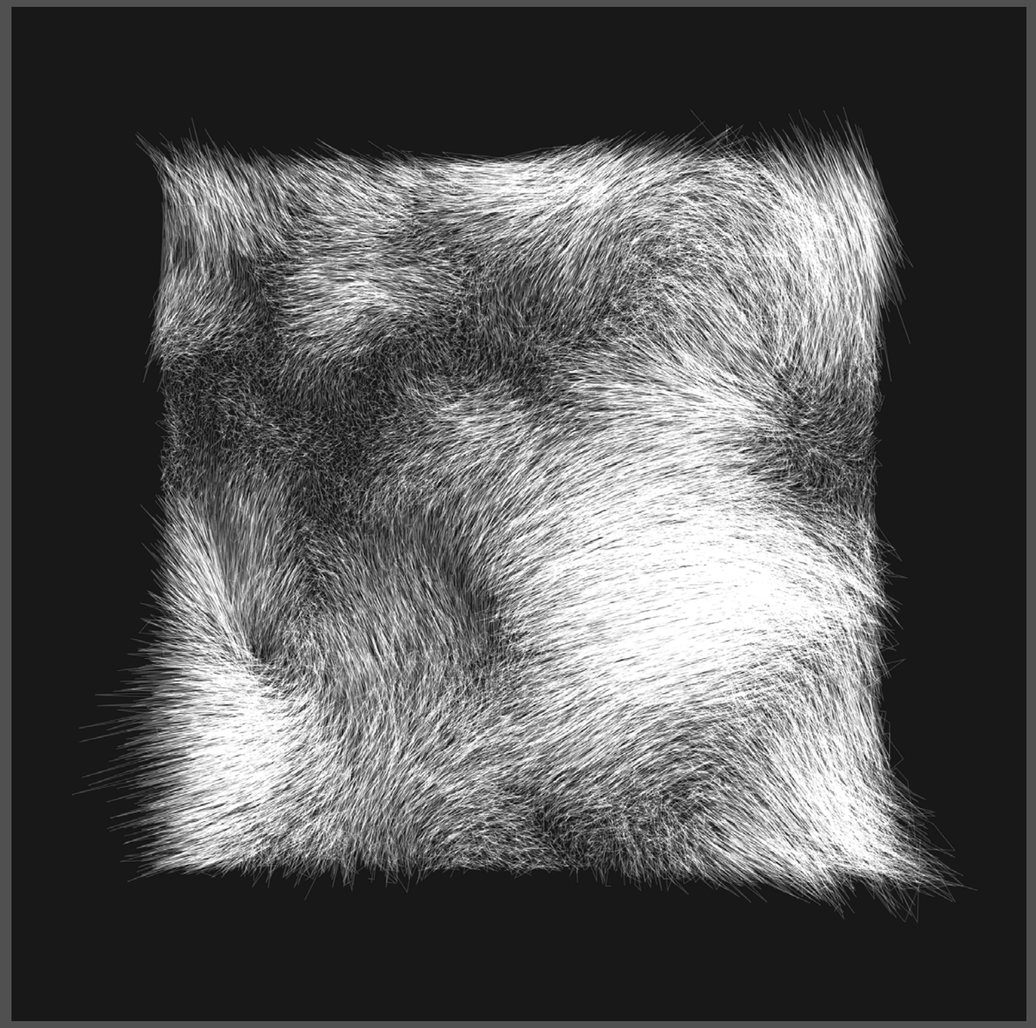


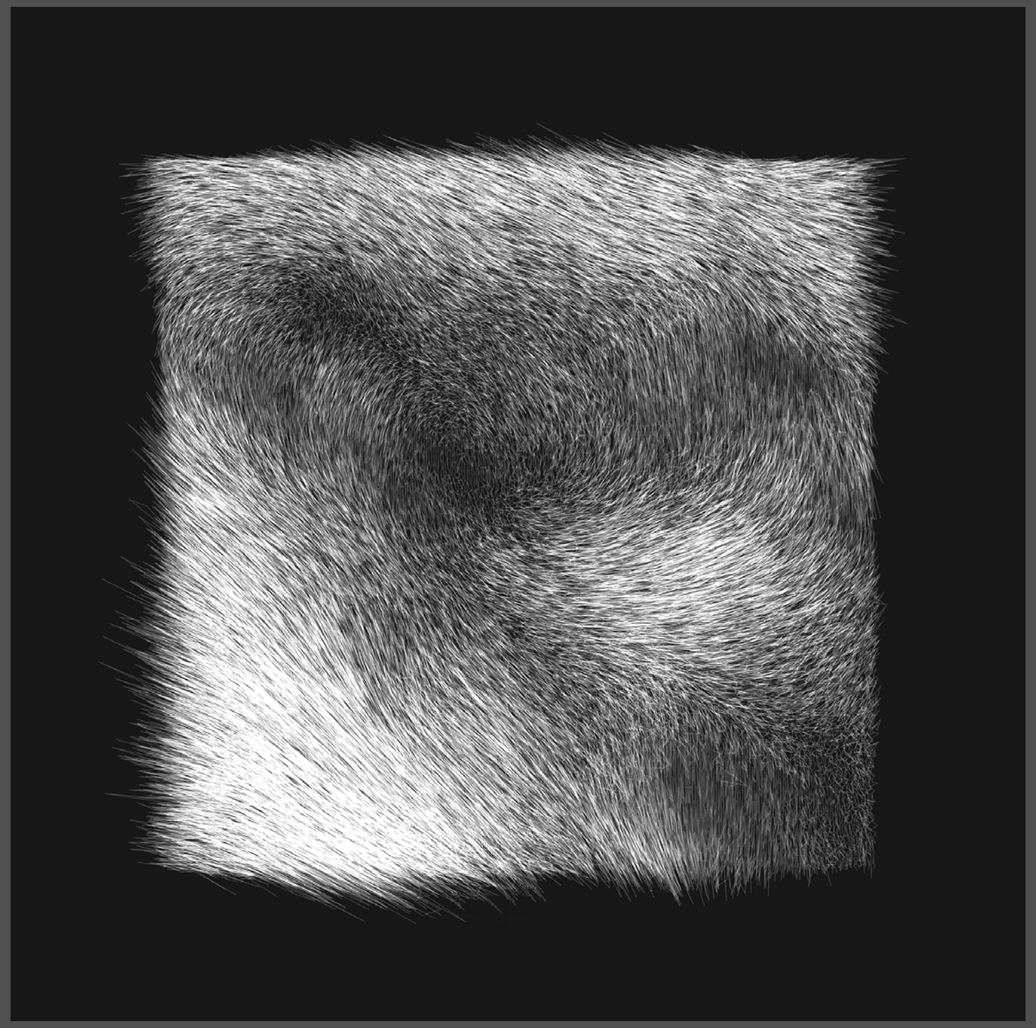


Konami

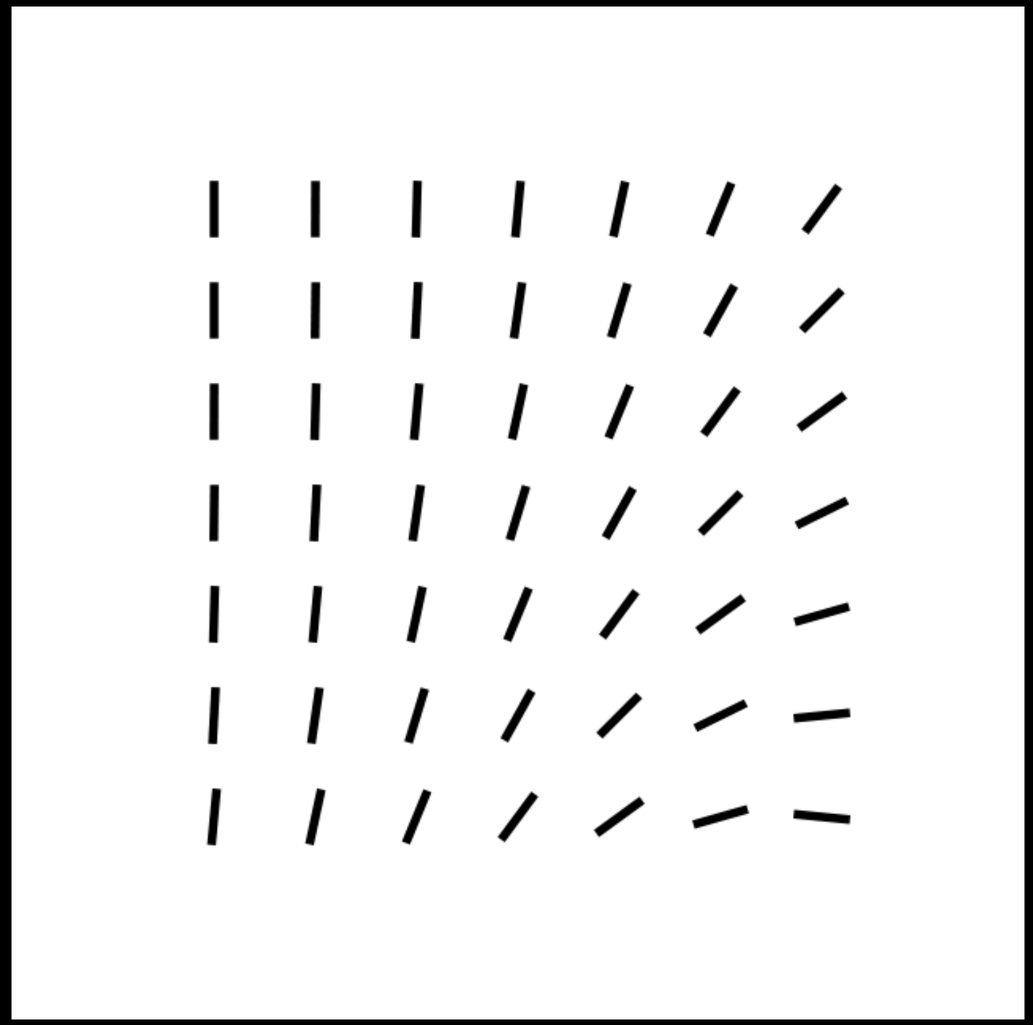
Unicode Compositions

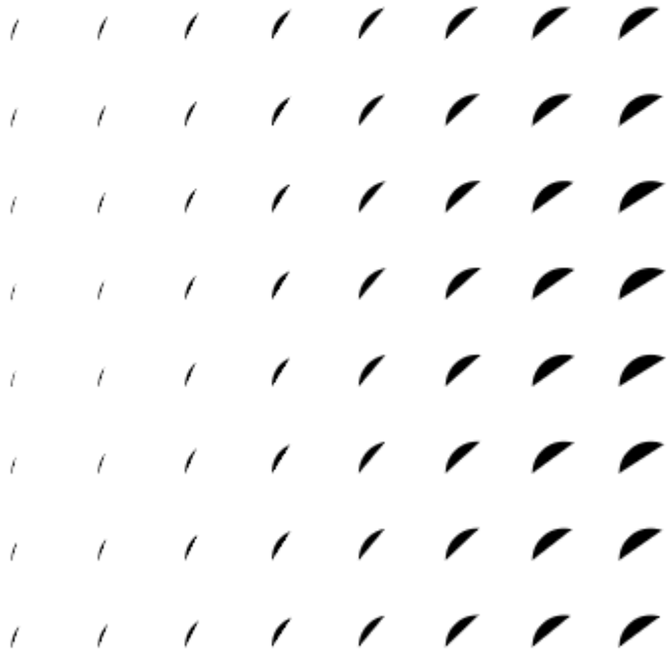
Unicode Compositions

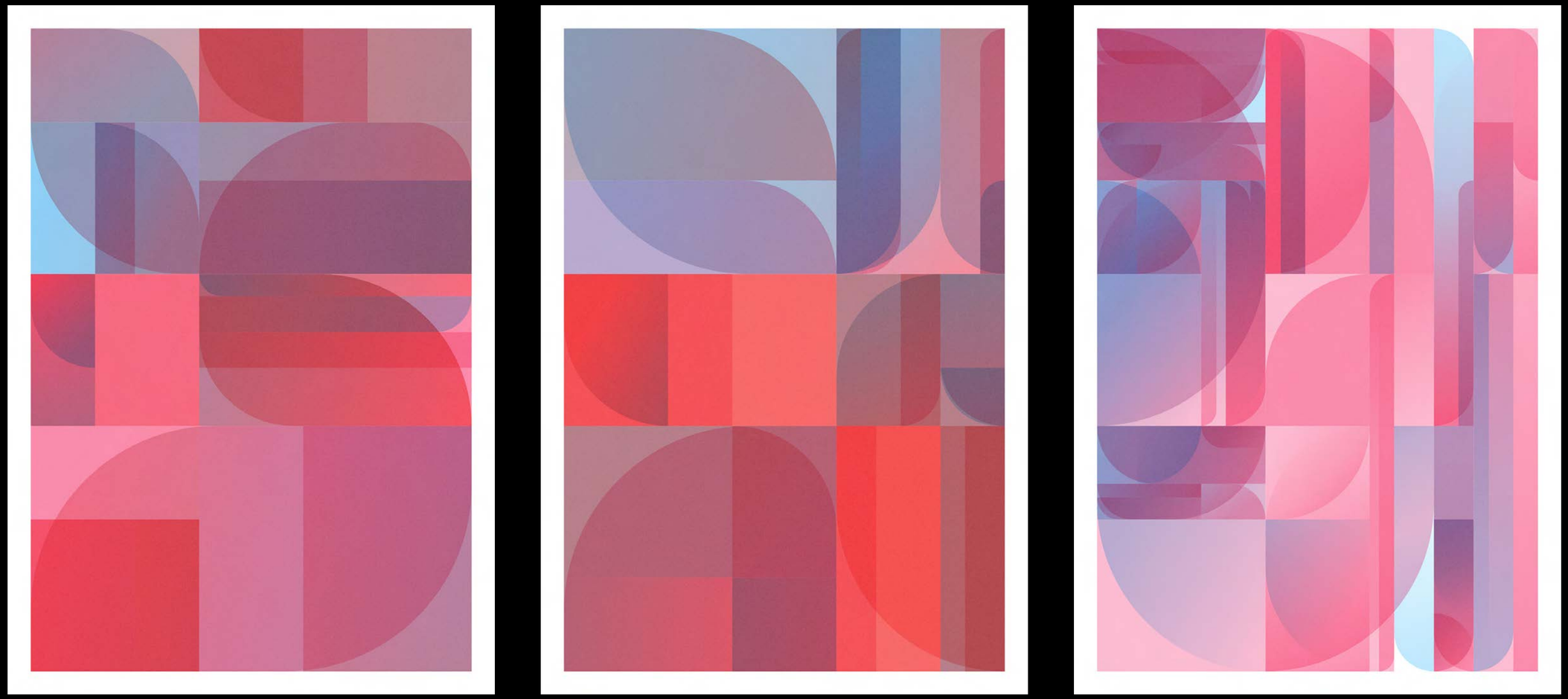




Computational Fur Rendering







TAU (2018)

TAU (2018)

time to code !

[take a breath]

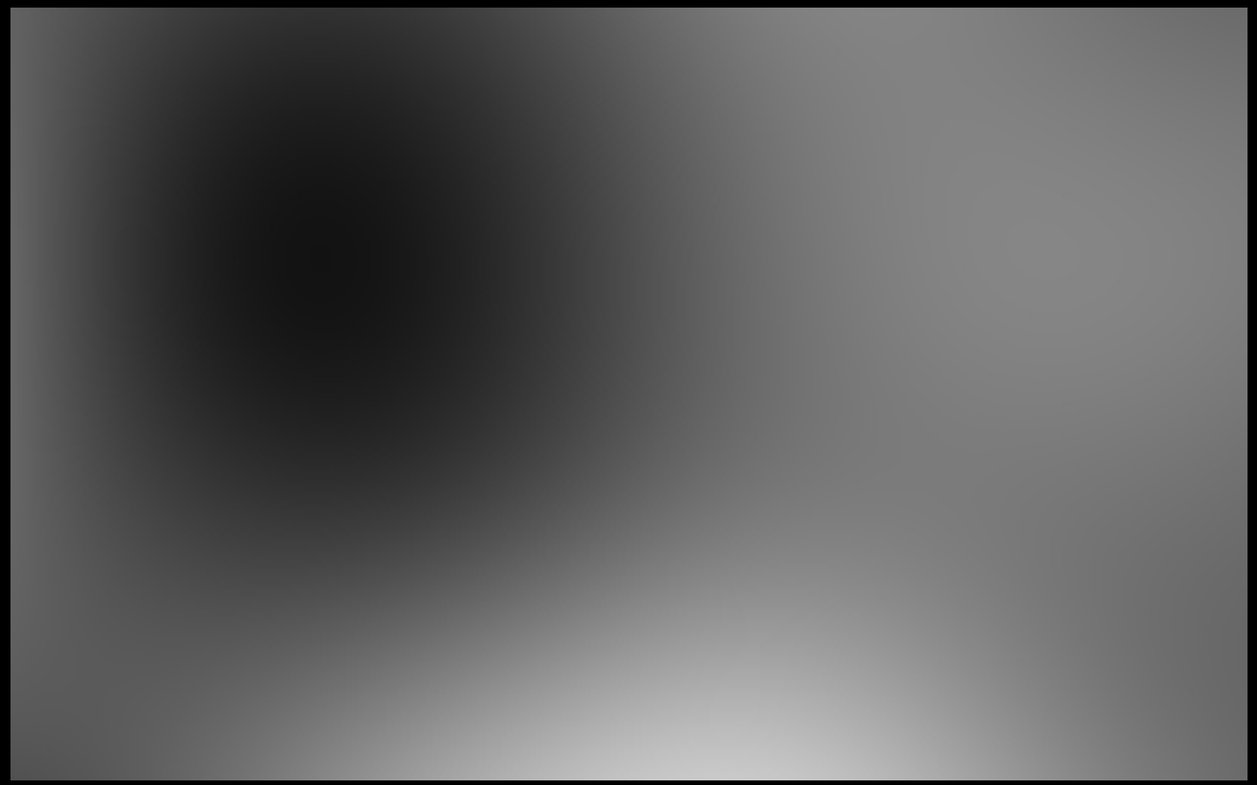
noise functions

```
v = noise2D(x, y)
v = noise3D(x, y, z)
v = noise4D(x, y, z, w)
```

```
// value is in -1...1 range
const v = noise2D(x, y);

// map to 0..1 range
const n = v * 0.5 + 0.5;

// turn into a percentage
const L = Math.floor(n * 100);

// get color value
const hsl = `hsl(0, 0%, ${L}%)`;
```

```
// frequency of the noise signal
const frequency = 5.0;

const v = noise2D(x * frequency, y * frequency);
```

frequency = 5.0



frequency = 0.5

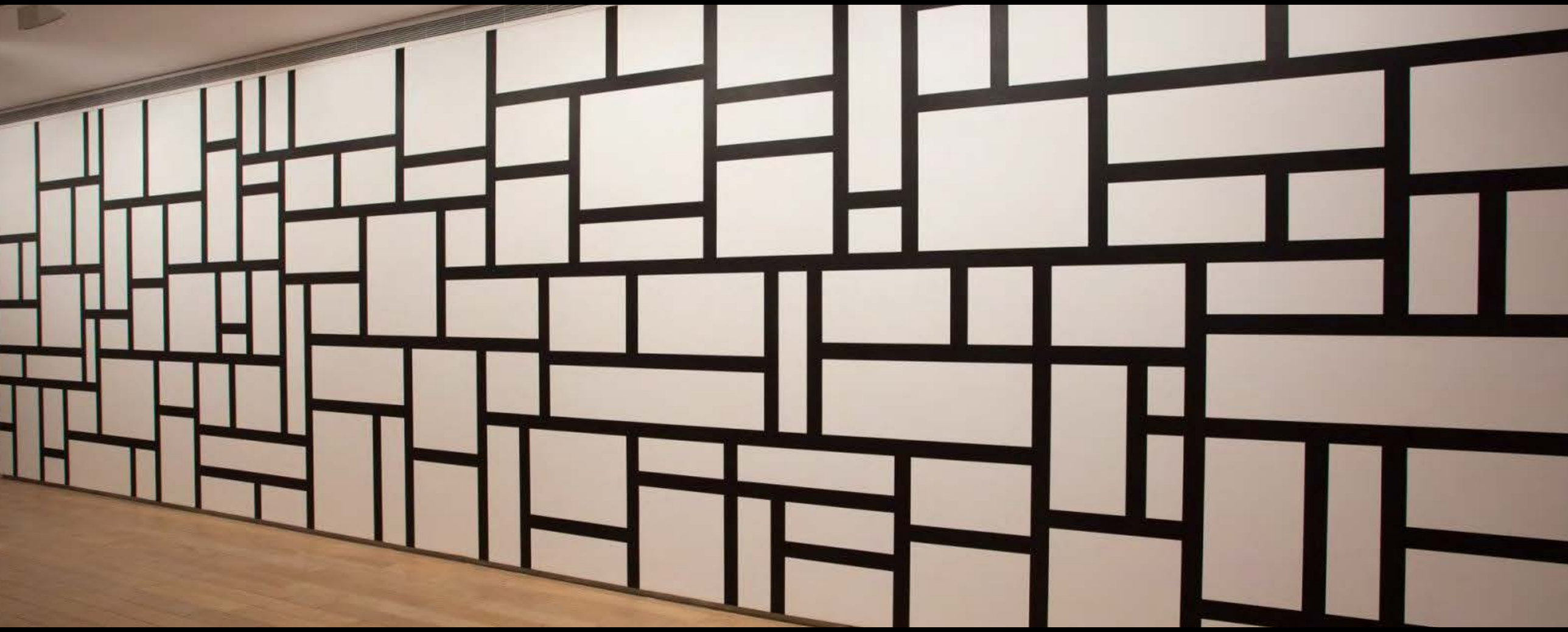
theory

Sol LeWitt (1928 – 2007)

Wall Drawing #1131

Wall Drawing #260

Wall Drawing #692



Wall Drawing #614

LeWitt created the following instructions for this work:

A six-inch (15 cm) grid covering the walls. Lines from corners, sides, and center of the walls to random points on the grid.

1st wall: Red lines from the midpoints of four sides;

2nd wall: Blue lines from four corners;

3rd wall: Yellow lines from the center;

4th wall: Red lines from the midpoints of four sides, blue lines from four corners;

5th wall: Red lines from the midpoints of four sides, yellow lines from the center;

6th wall: Blue lines from four corners, yellow lines from the center;

7th wall: Red lines from the midpoints of four sides, blue lines from four corners, yellow lines from the center.

Each wall has an equal number of lines. (The number of lines and their length are determined by the draftsman.)

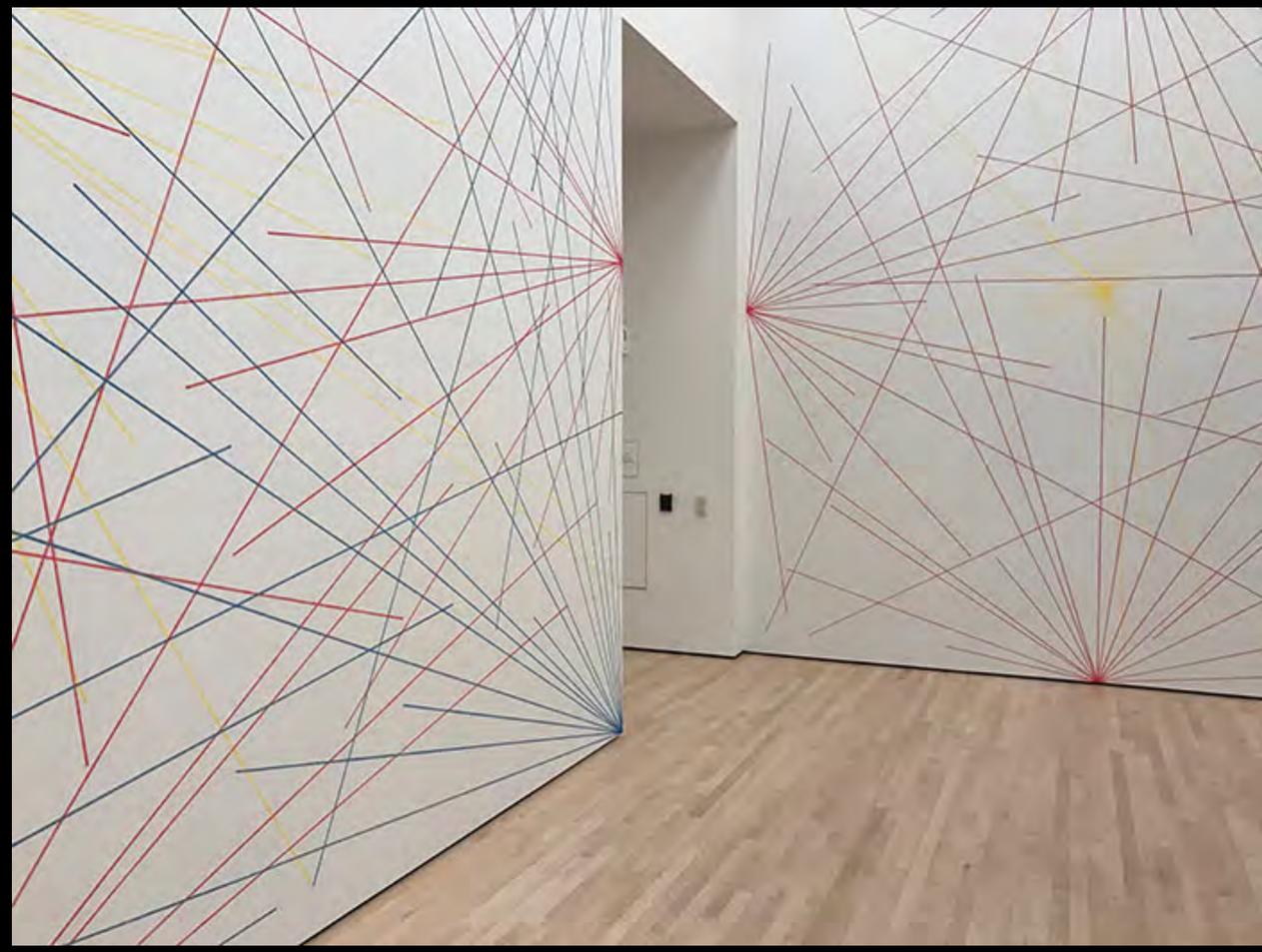
Artist Roland Lusk on making LeWitt's *Wall Drawings*

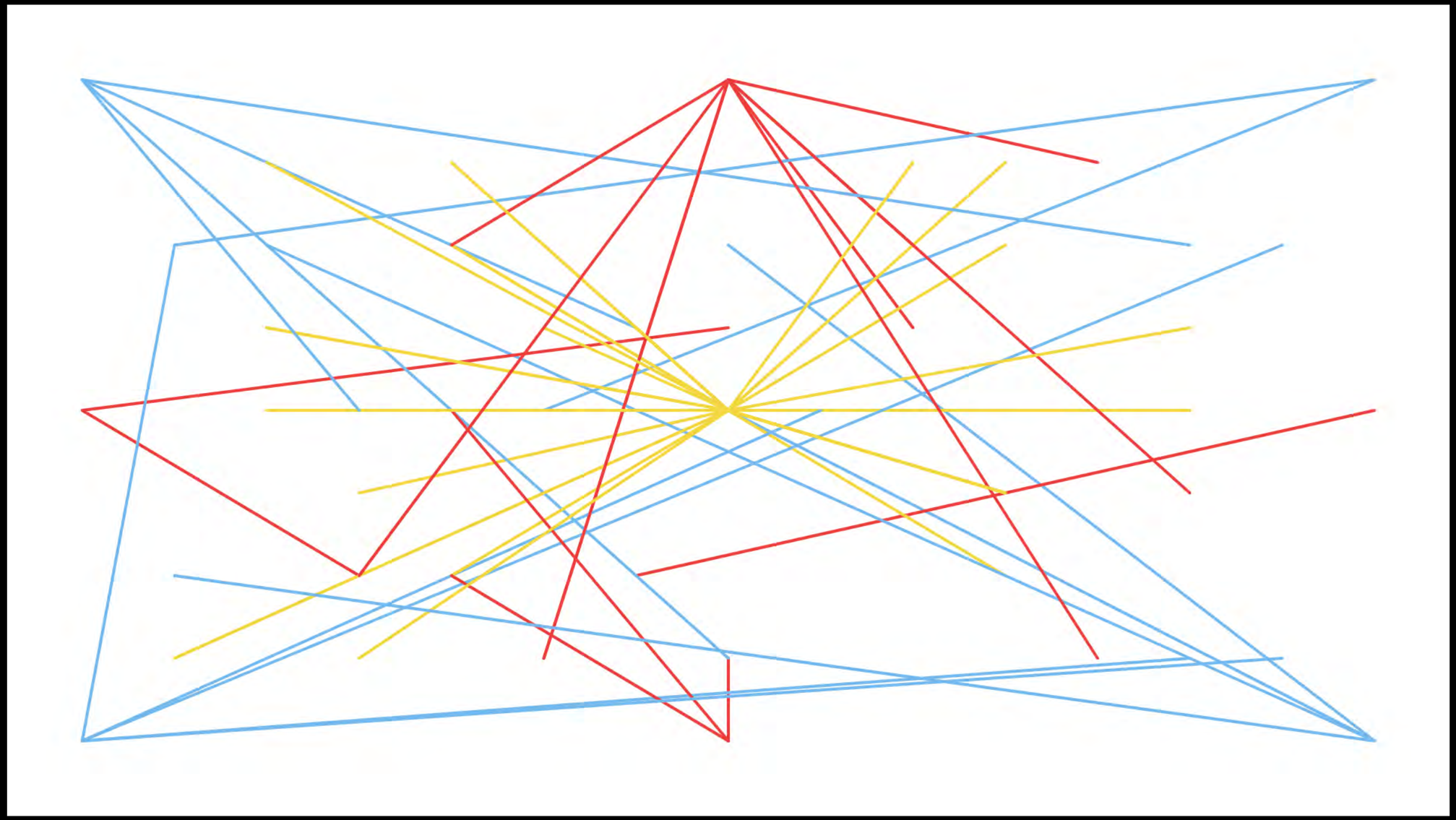
Wall Drawing #273

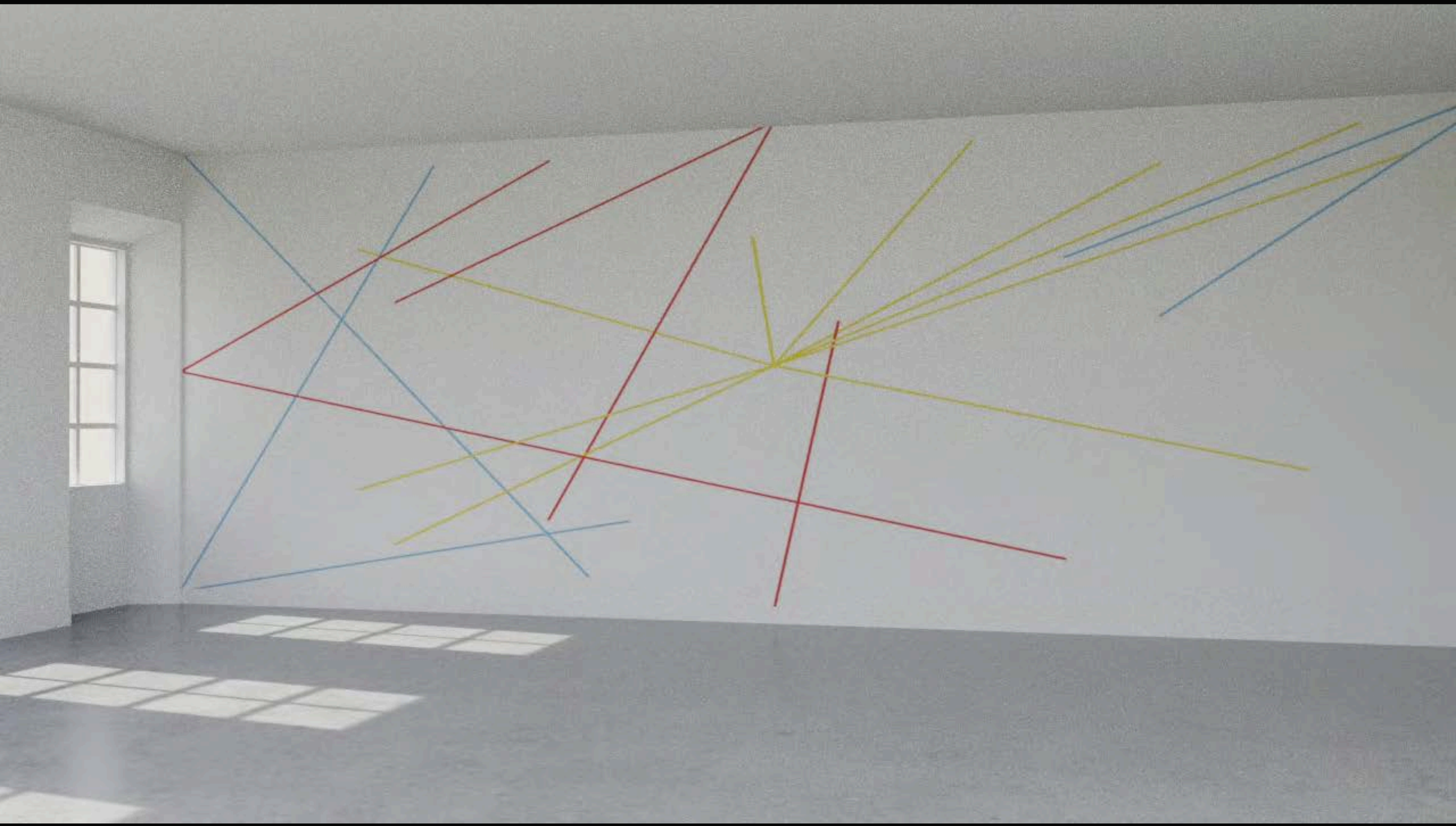
A six-inch (15 cm) grid covering the walls. Lines from corners, sides, and centre of the walls to random points on the grid.

...

7th wall: Red lines from the midpoints of four sides, blue lines from four corners, yellow lines from the center.



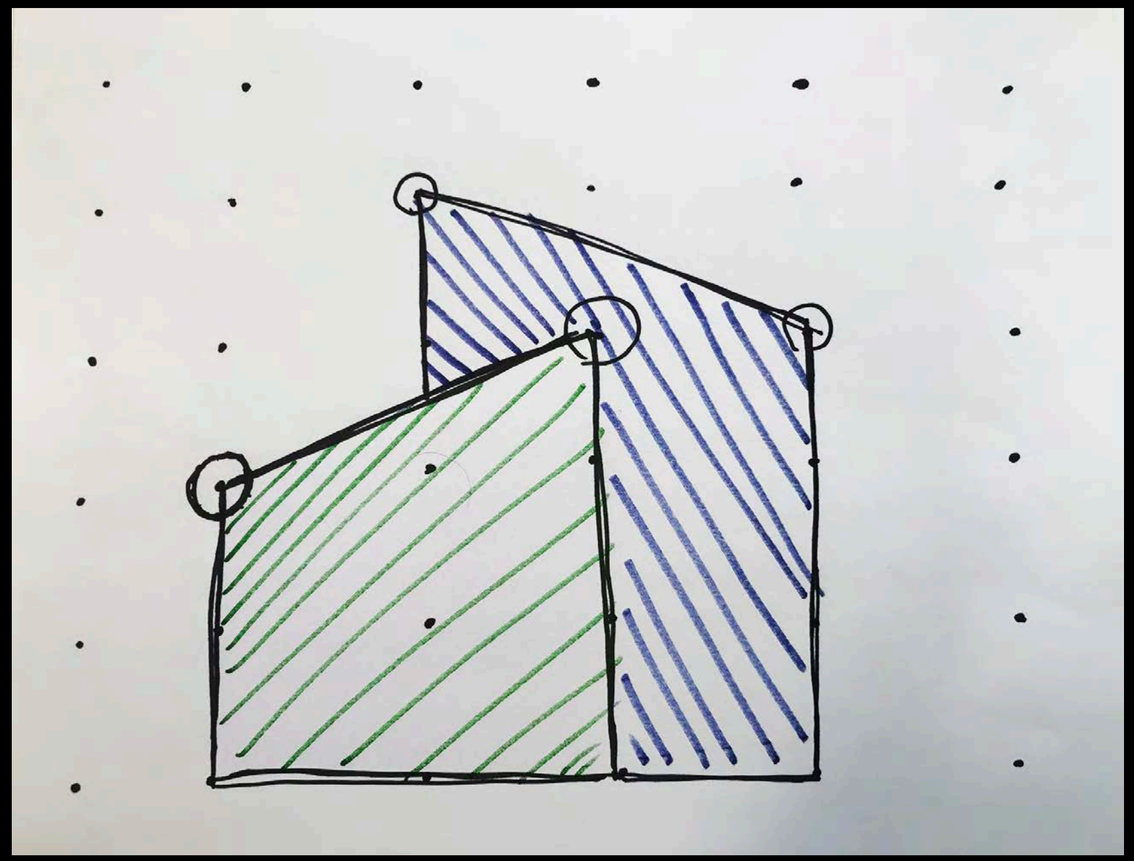




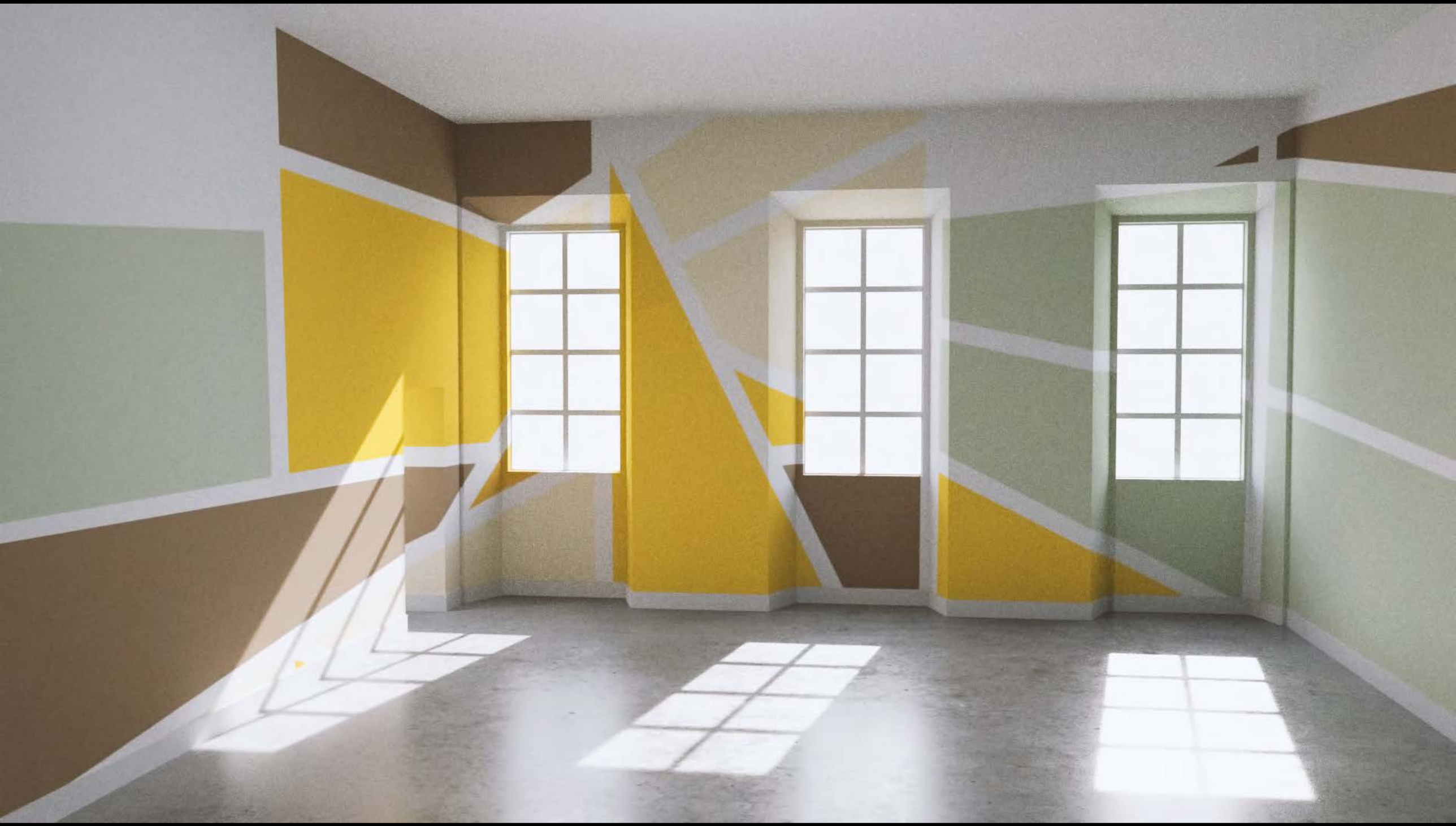
https://solvingsol.com/

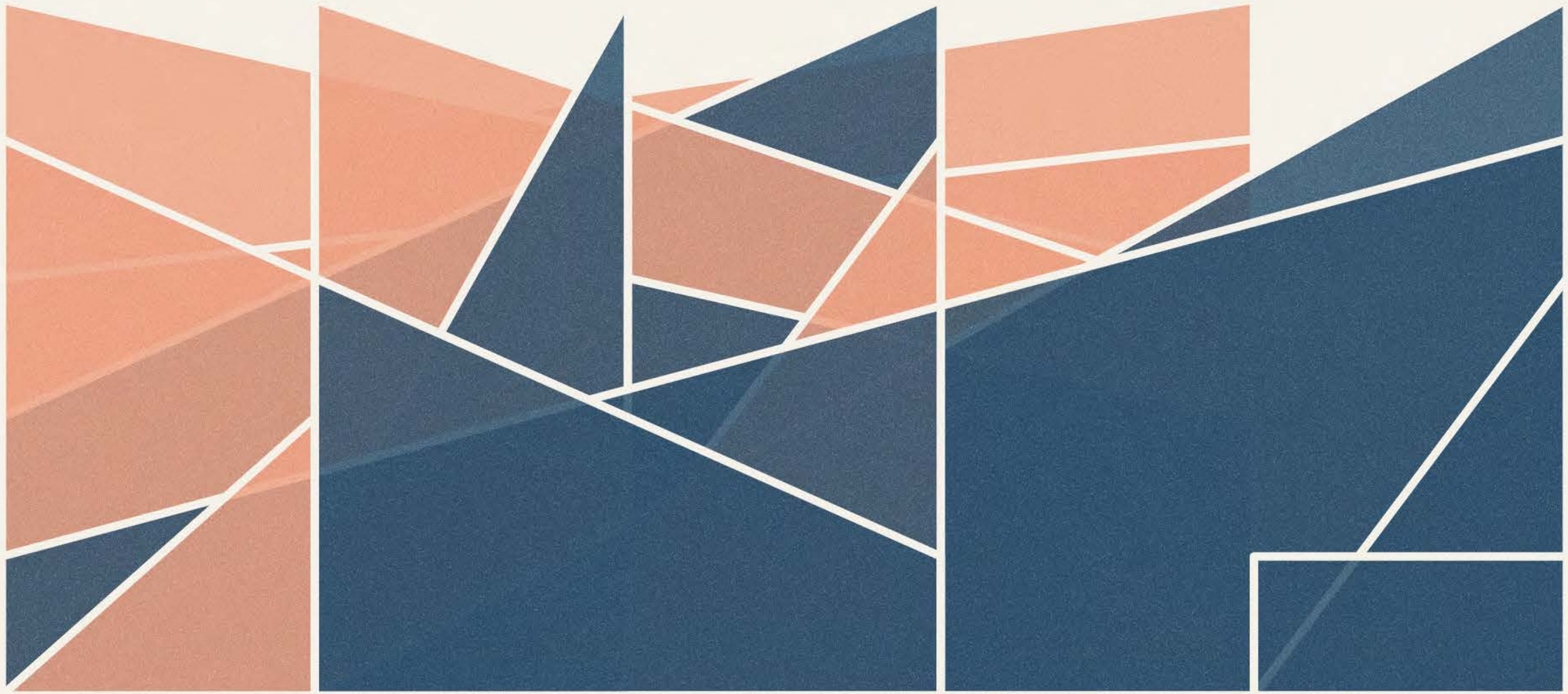
Generative Wall Drawing #2

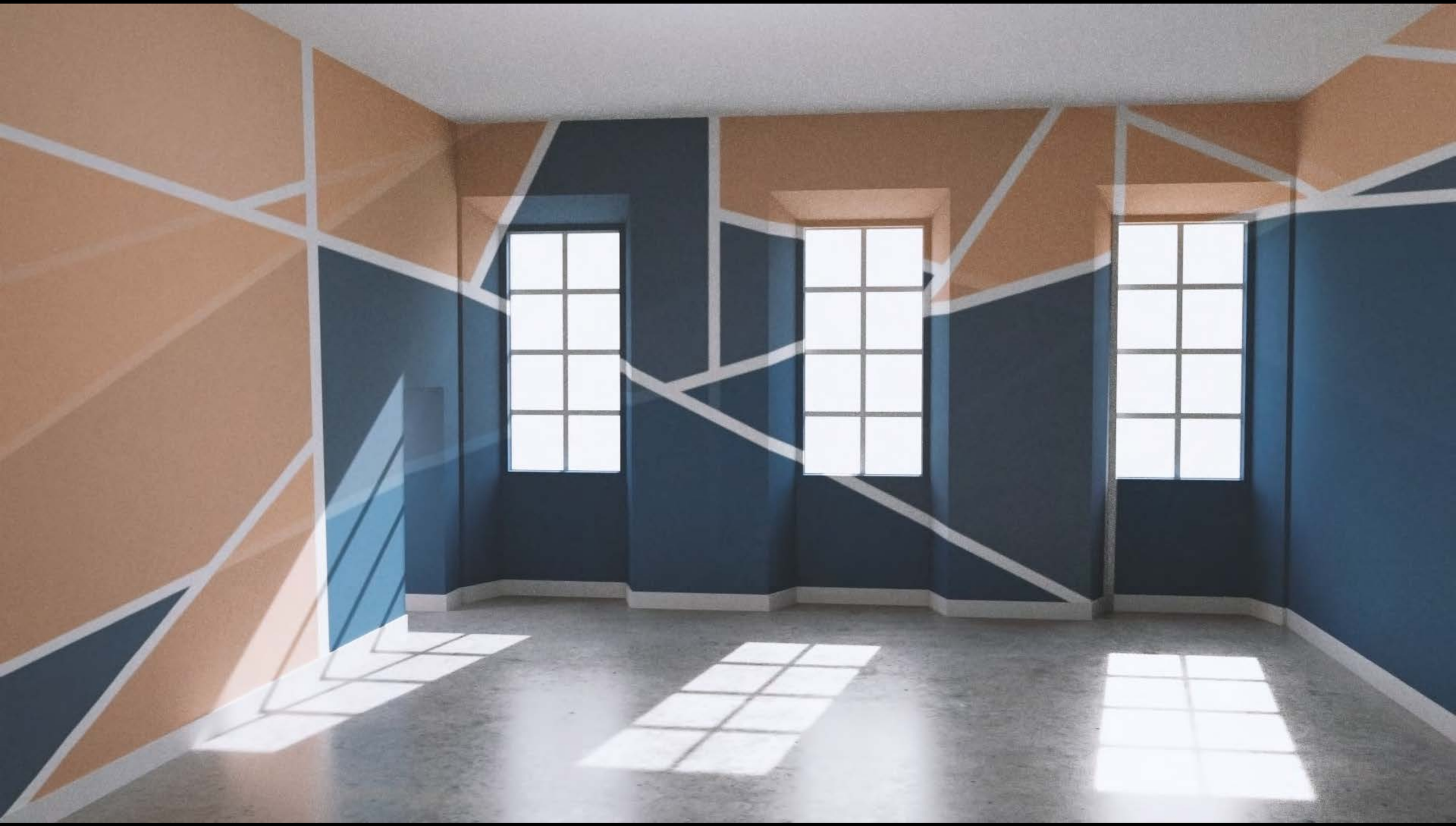
- Using a 6x6 grid of evenly spaced points
- Connect two random points on the grid; forming a trapezoid with two parallel sides extending down
- Fill the trapezoid with a colour, then stroke with the background colour
- Find another two random points and repeat; continuing until all grid points are exhausted
- Layer the shapes by the average Y position of their two grid points



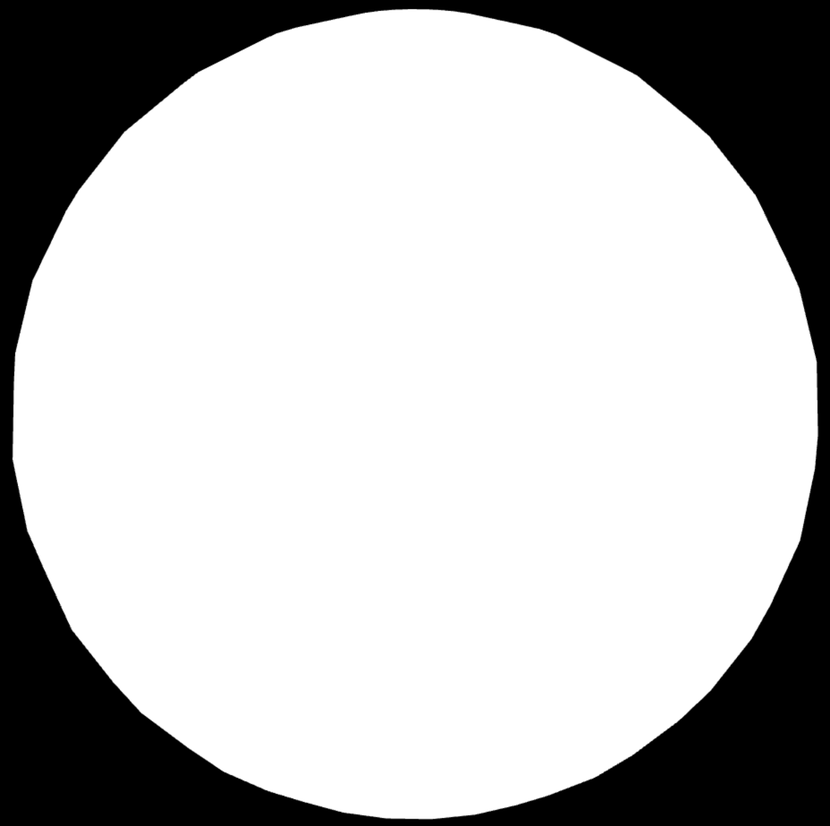
welcome back !





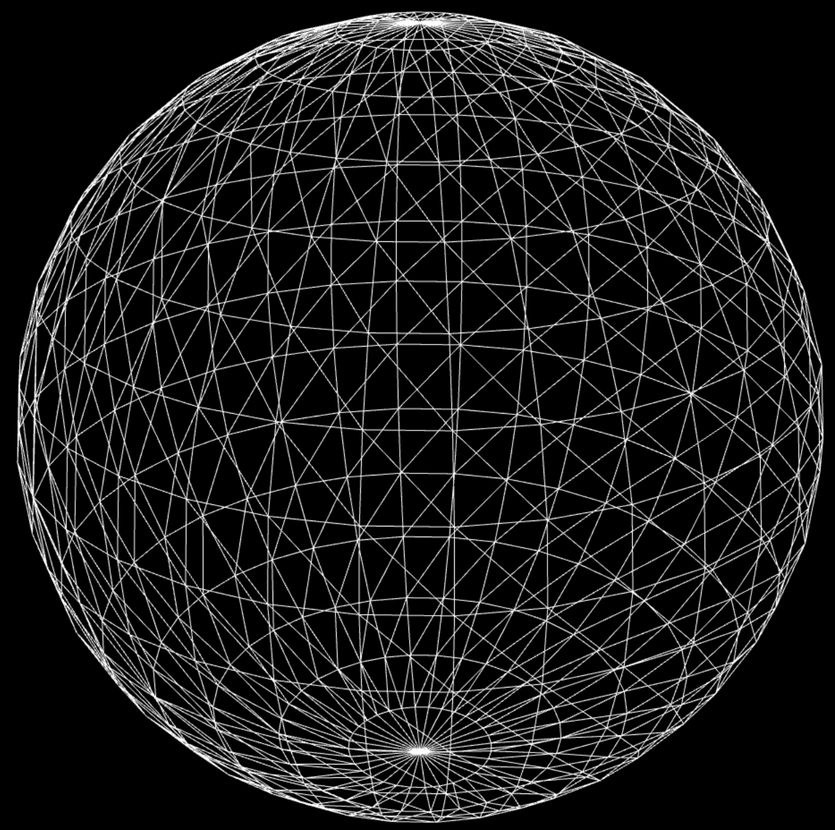


the third dimension



material

MeshBasicMaterial

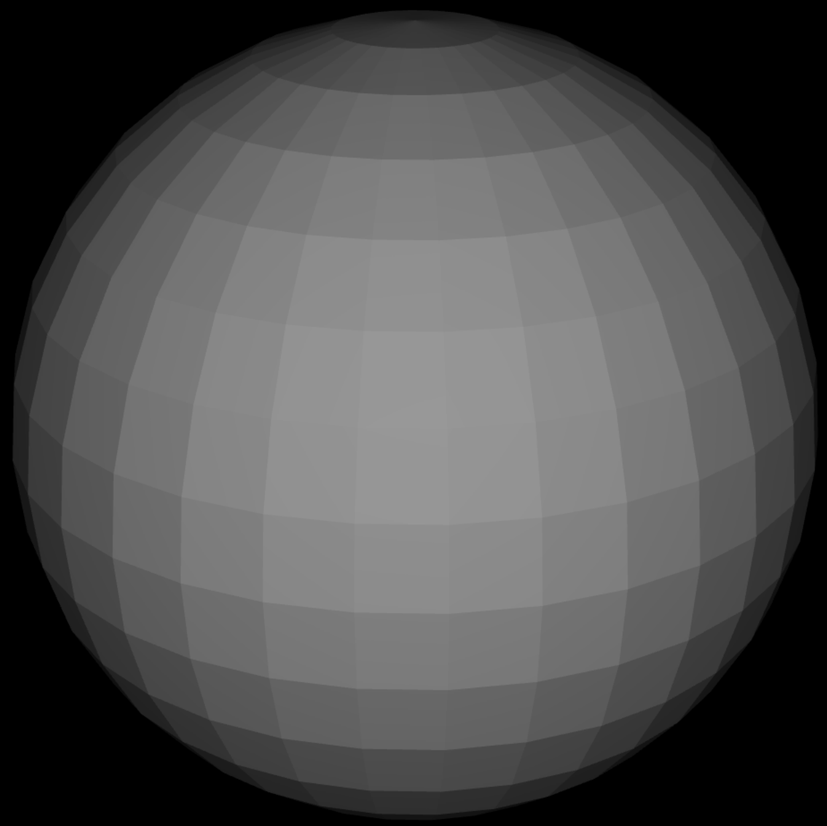


material

MeshBasicMaterial (wireframe)

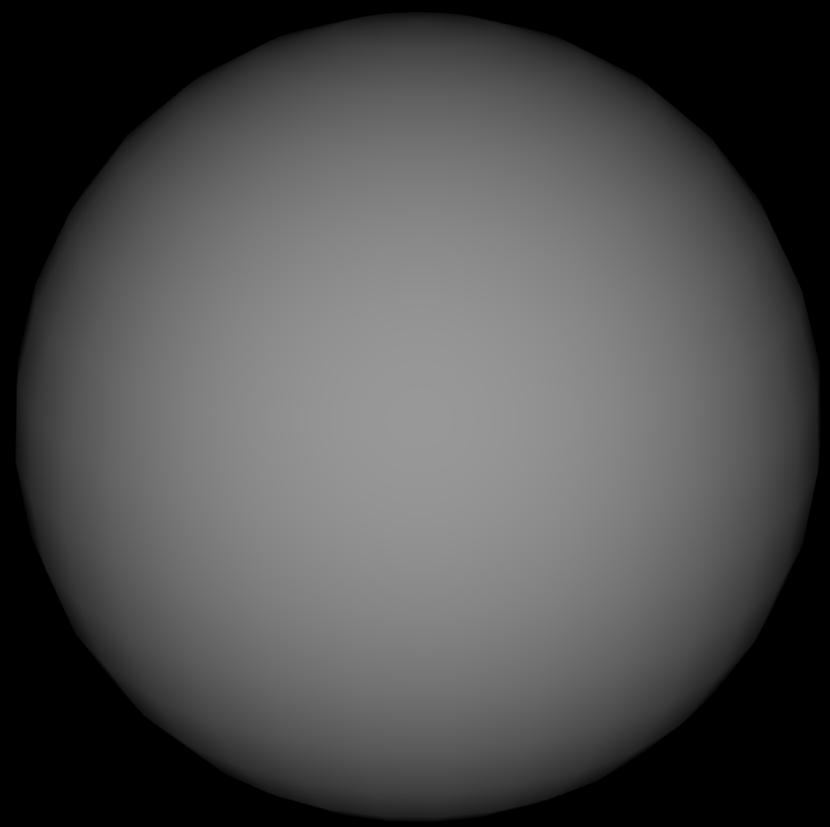
material

MeshNormalMaterial



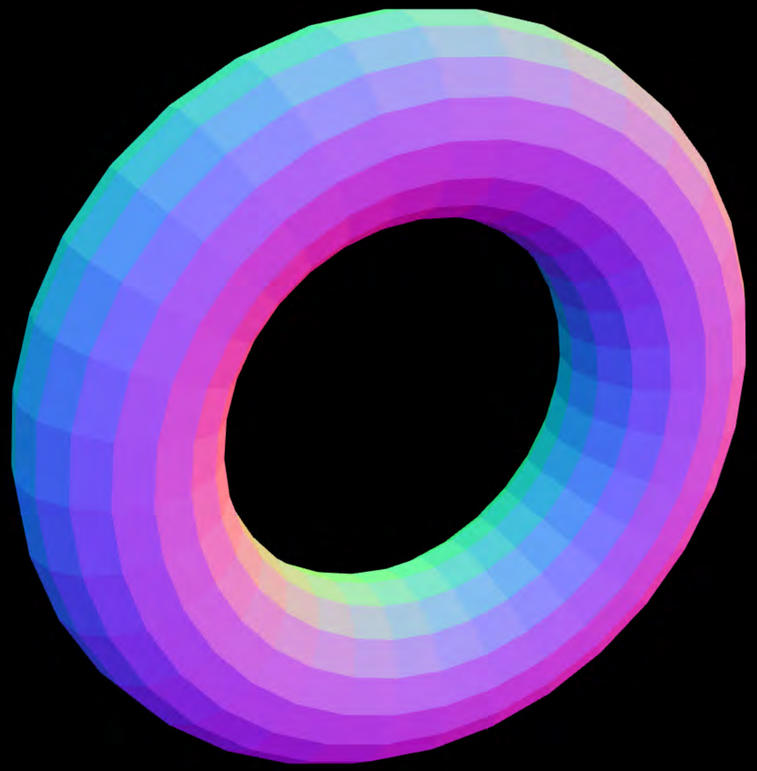
material

MeshStandardMaterial



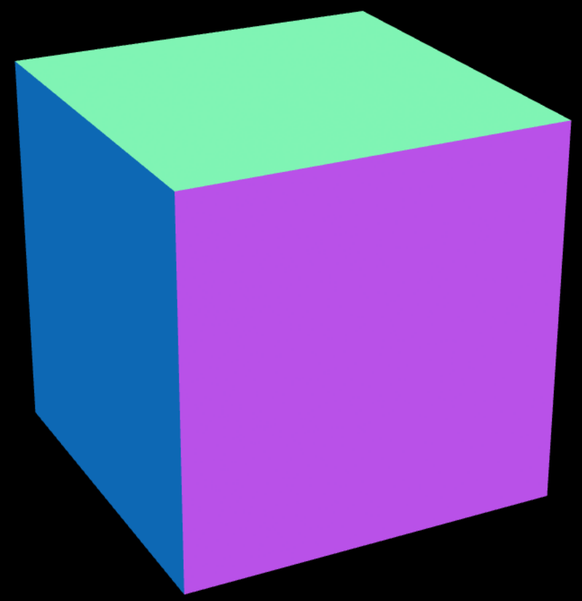
material

MeshStandardMaterial



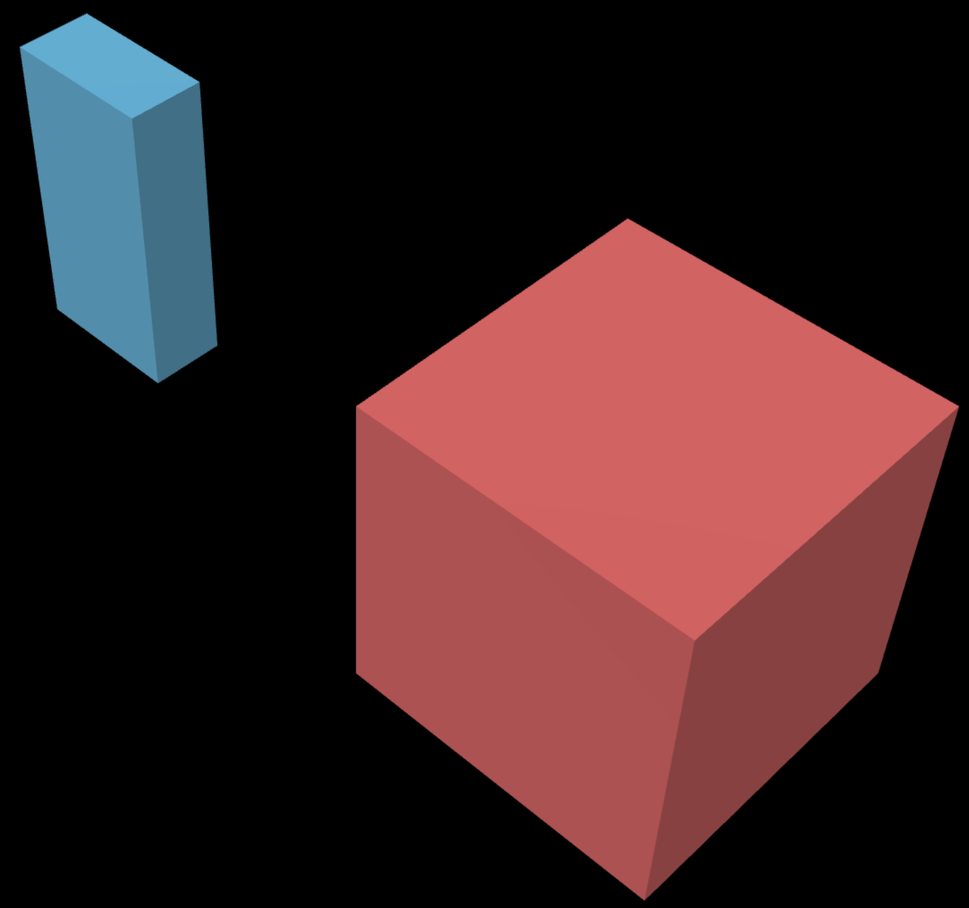
geometry

TorusGeometry

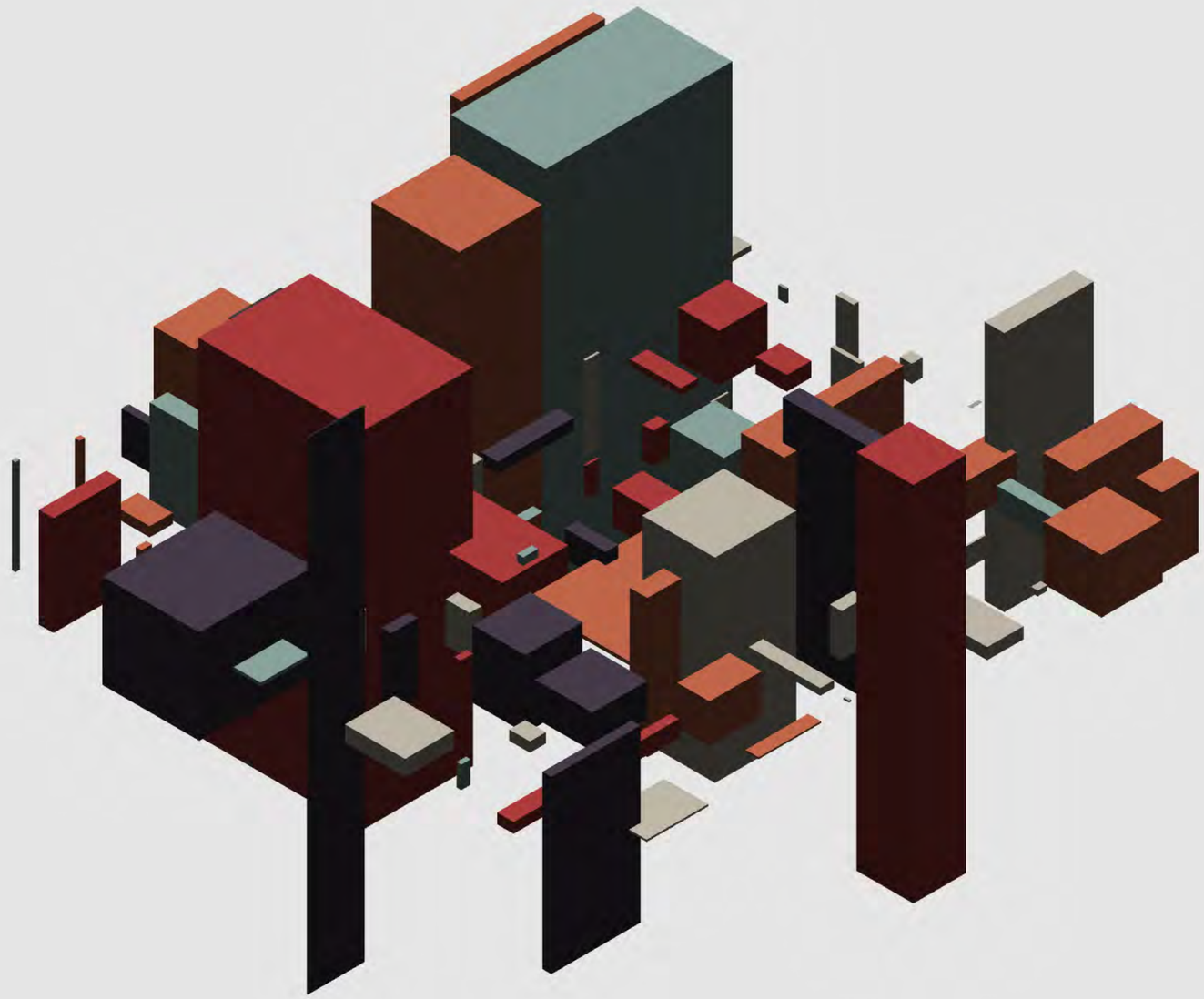


geometry

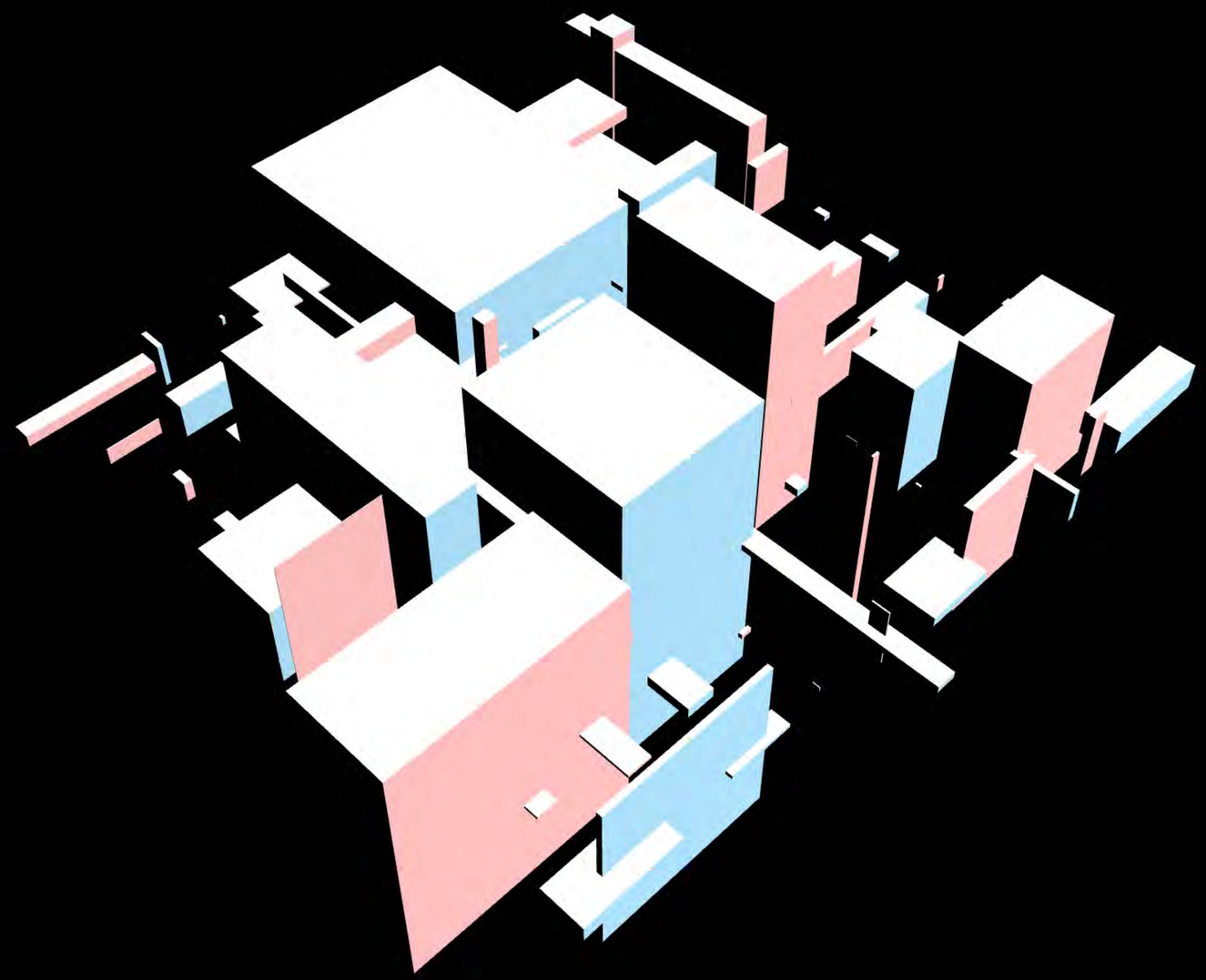
BoxGeometry



mesh

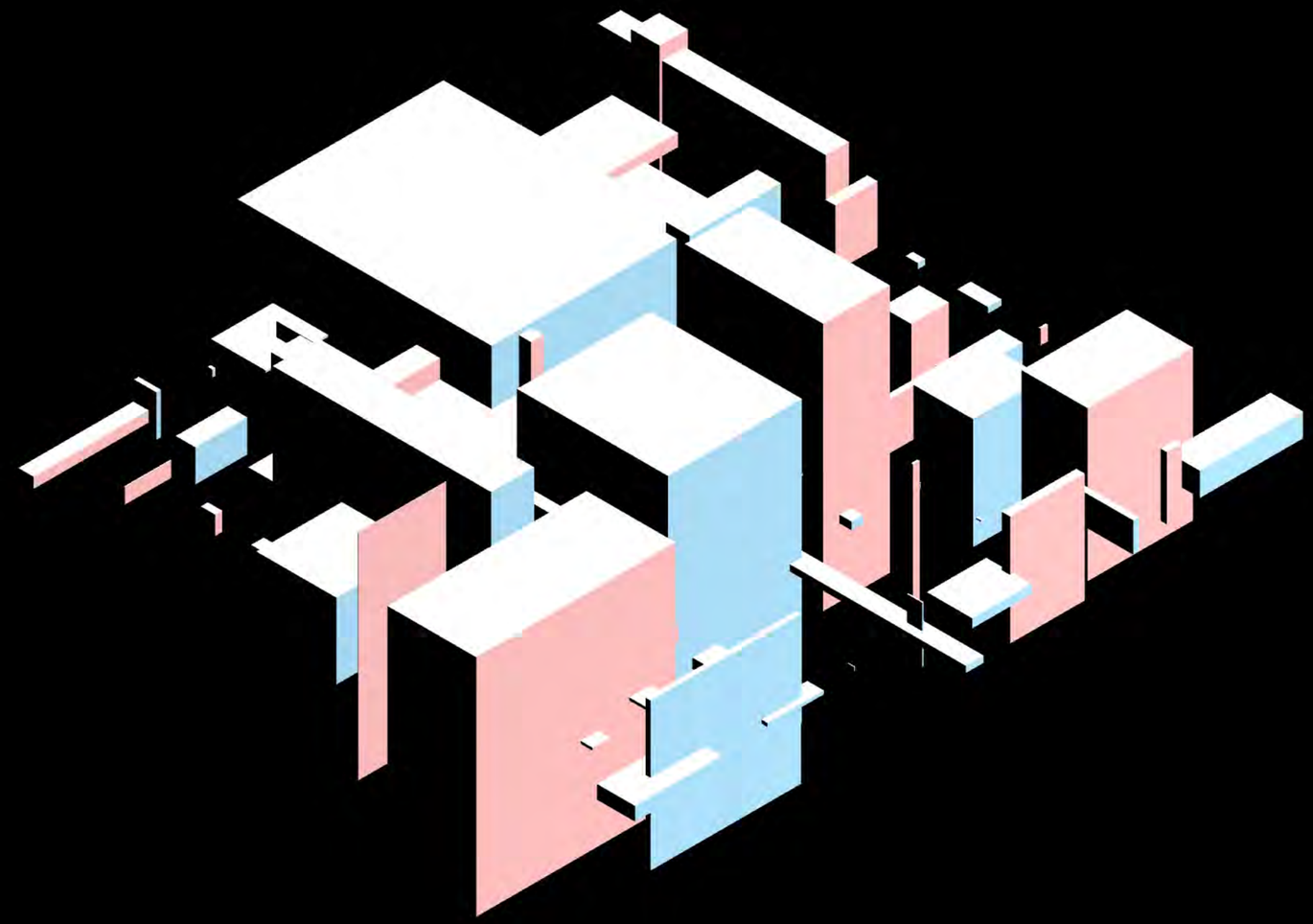


scene



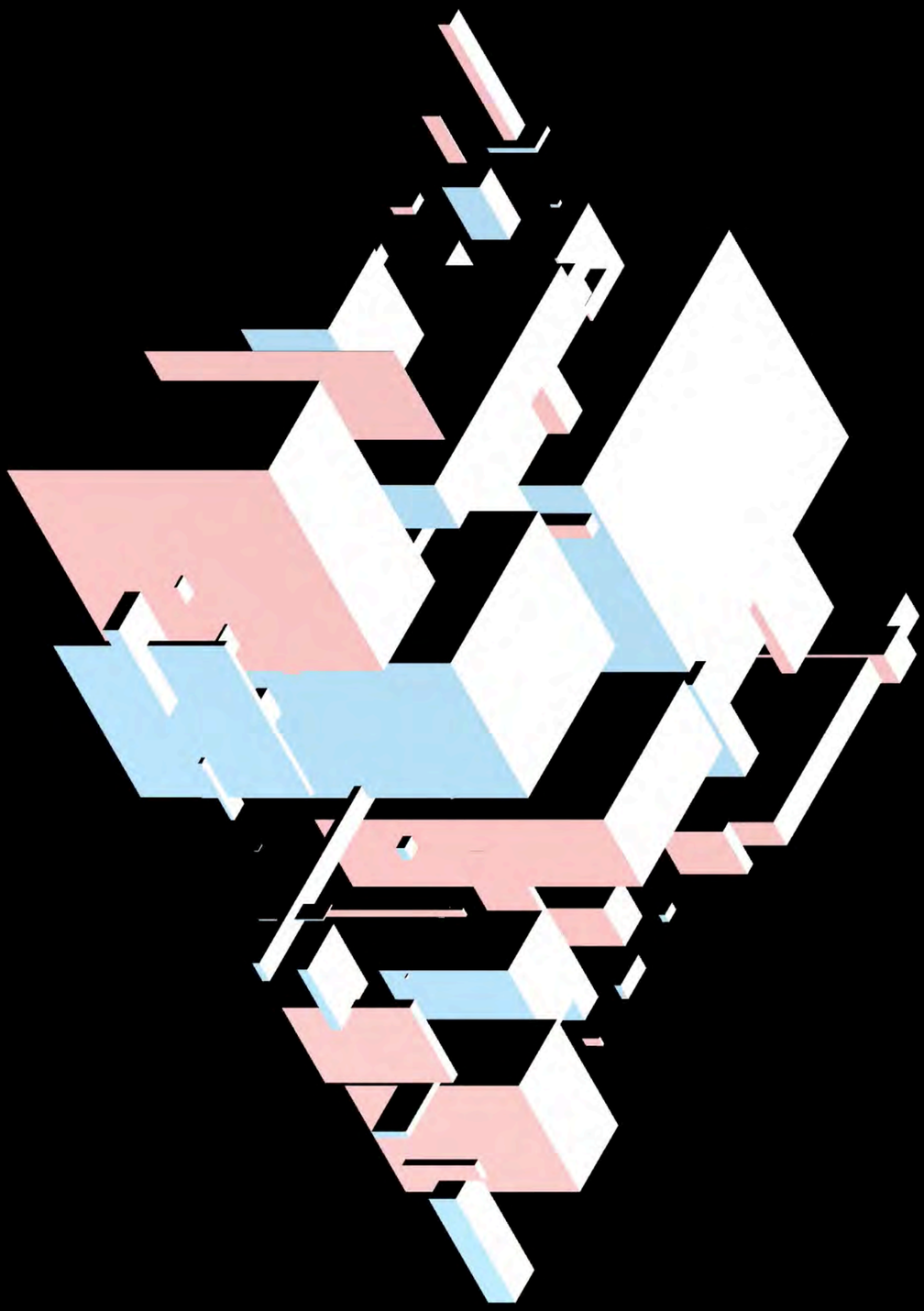
camera

PerspectiveCamera



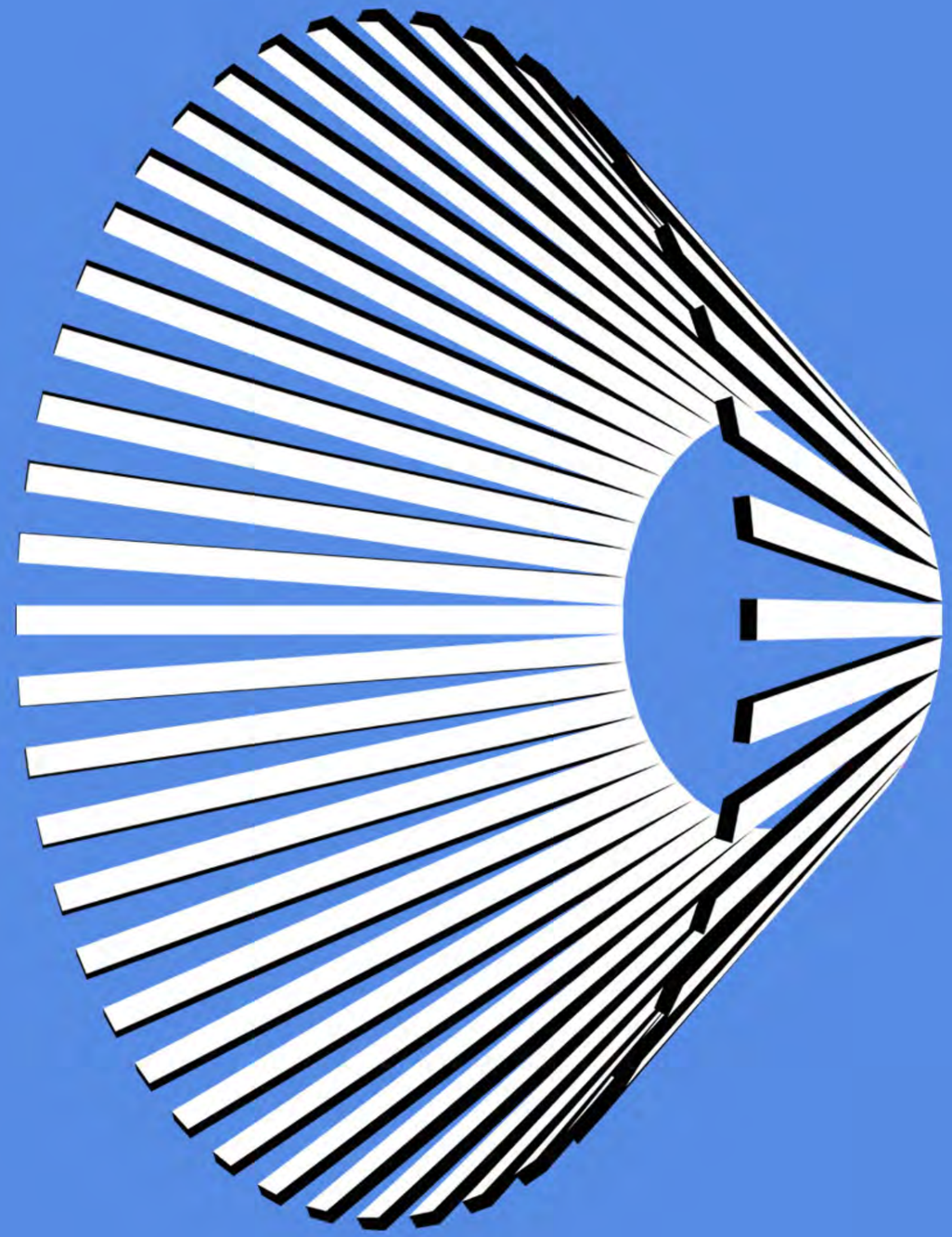
camera

OrthographicCamera



time to code !

questions so far ?



44º Sesc Melhores Filmes

44º Sesc Melhores Filmes

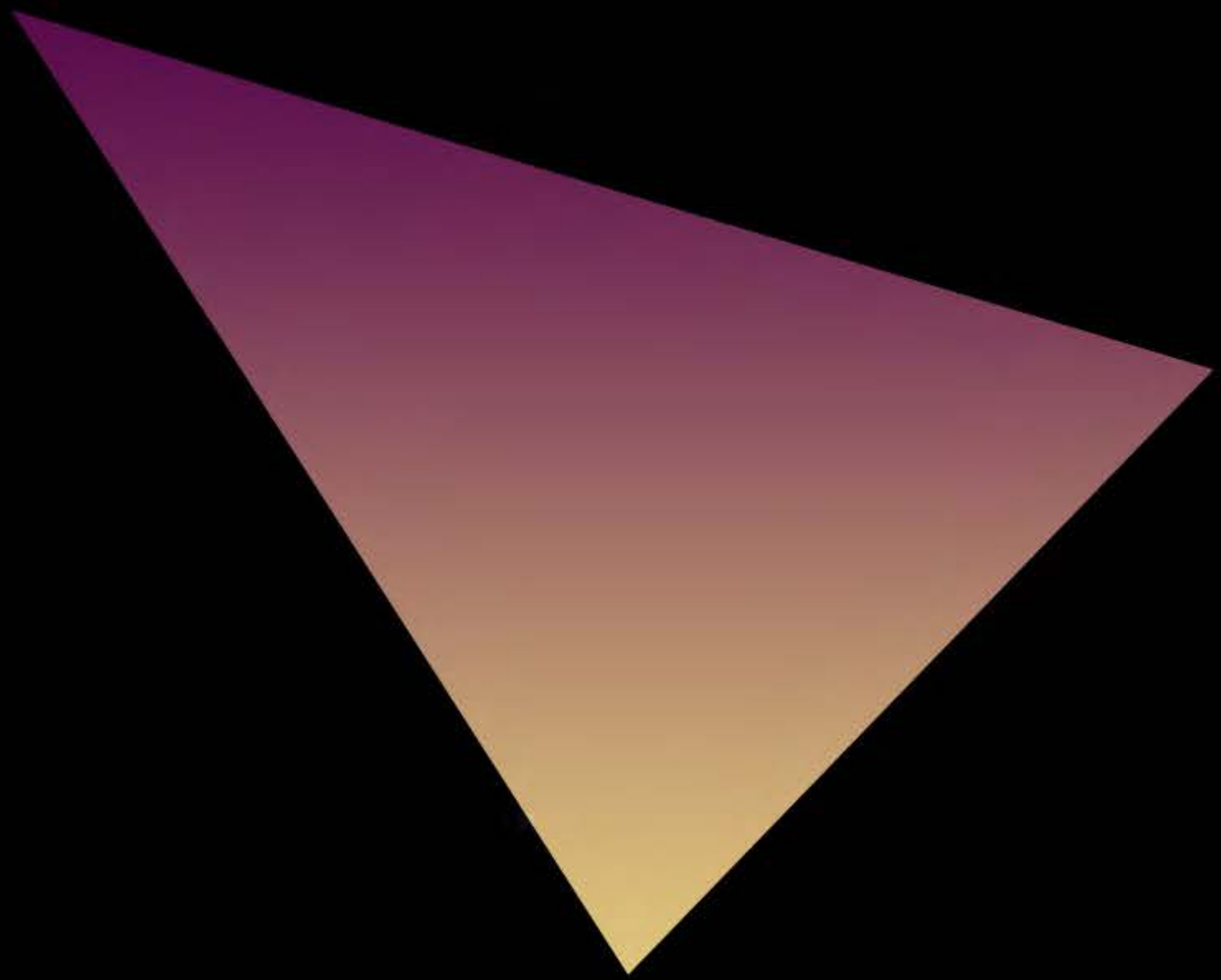

44º FESTIVAL SESC MELHORES FILMES
2018
sesc
05 a 25 abril
Programação
Sumário
PREMIADOS DO ANO
FILMES NACIONAIS PREMIADOS
FILMES NACIONAIS INDICADOS
FILMES ESTRANGEIROS PREMIADOS
FILMES ESTRANGEIROS INDICADOS
CLÁSSICOS RESTAURADOS EM 4K
PROGRAMAÇÃO
ATIVIDADES FORMATIVAS
44º FESTIVAL SESC MELHORES FILMES
2018
44º FESTIVAL SESC MELHORES FILMES

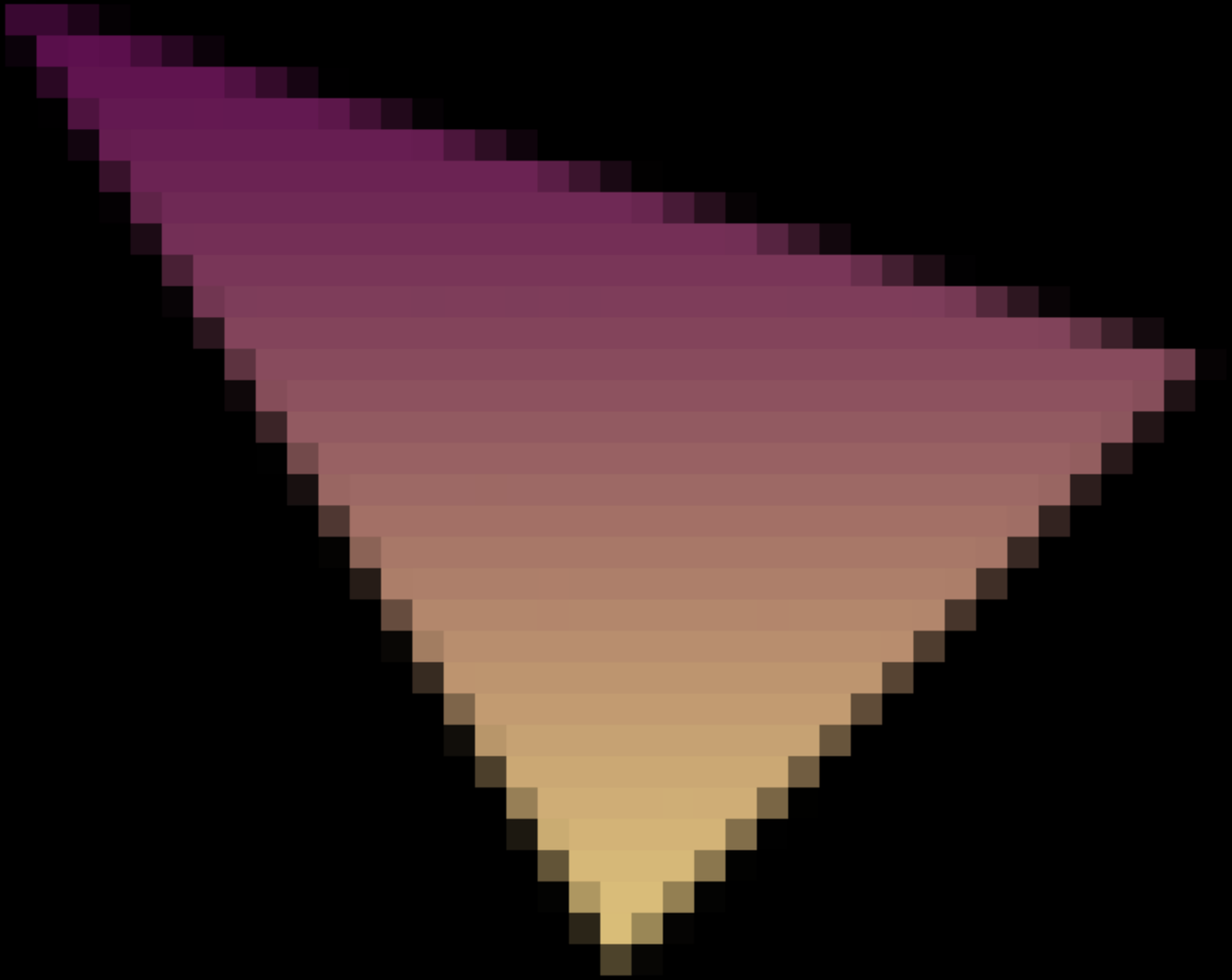
let’s make a GIF !

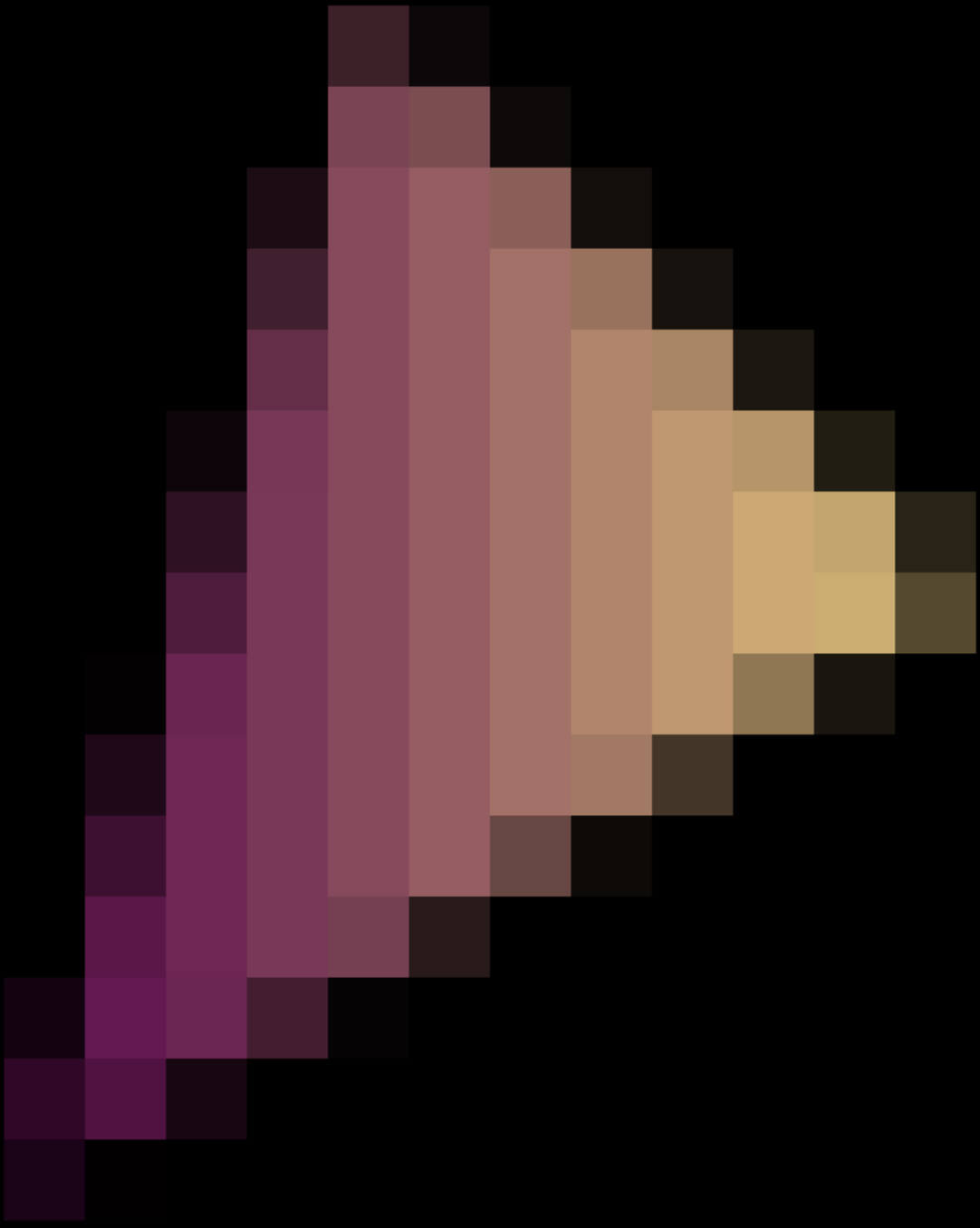
https://giftool.surge.sh/

shaders

```
// Default floating point precision
precision highp float;

// Inputs
varying vec2 vUv;

// Variables from JavaScript
uniform float time;

// Main function
void main () {
  // Output color
  gl_FragColor = vec4(1.0);
}
```

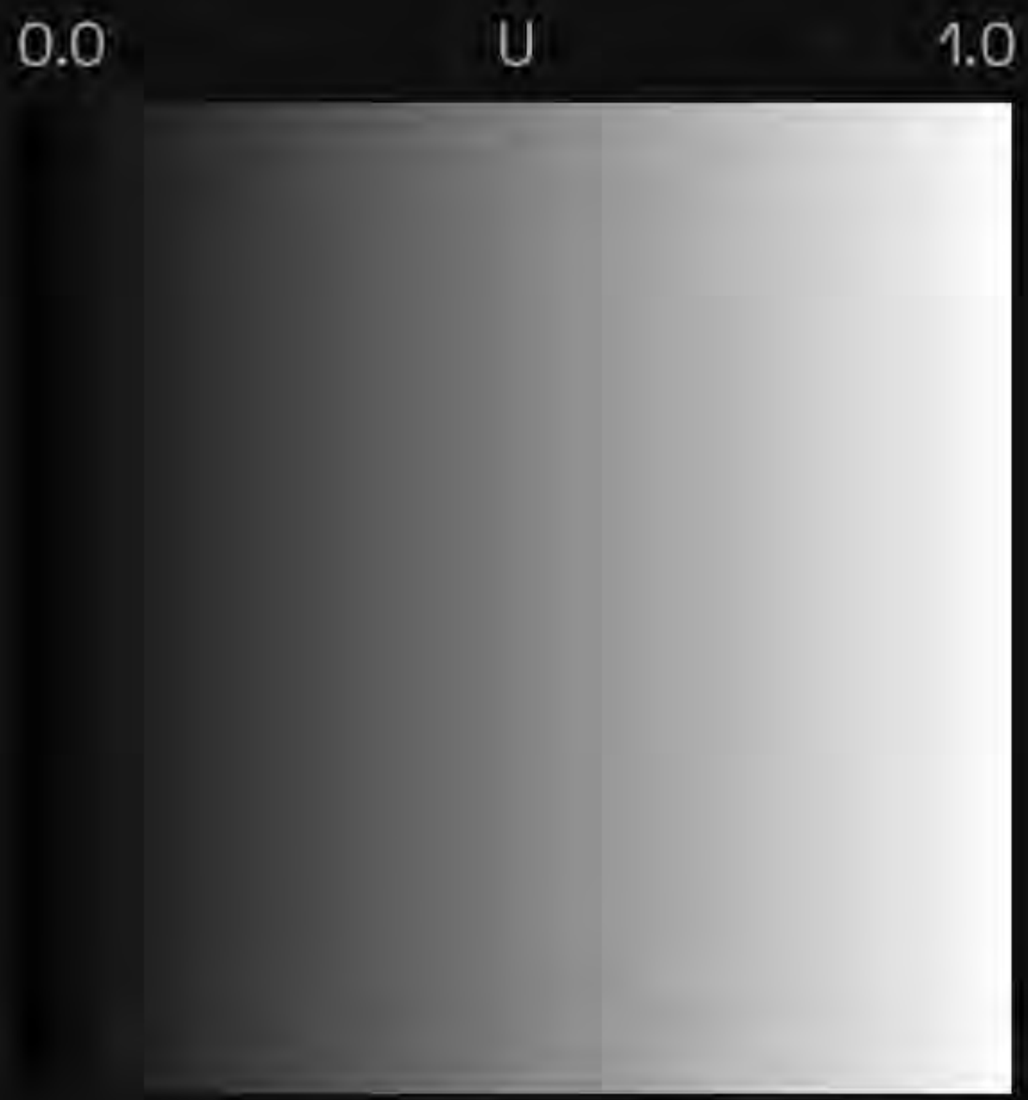






```
// Default floating point precision
precision highp float;

// Inputs
varying vec2 vUv;

// Variables from JavaScript
uniform float time;

// Main function
void main () {
  // Output color
  gl_FragColor = vec4(1.0);
}
```

```
precision highp float;

varying vec2 vUv;

// Main function
void main () {
  // Create a RGB color
  vec3 color = vec3(vUv.x);

  // Create an opacity
  float alpha = 1.0;

  // Output color
  gl_FragColor = vec4(color, alpha);
}
```




```
gl_FragColor = vec4(vec3(vUv.x), 1.0);
```


```
gl_FragColor = vec4(vec3(vUv.y), 1.0);
```

Primary Branding by DIA Studio

Primary Branding by DIA Studio

time to code !