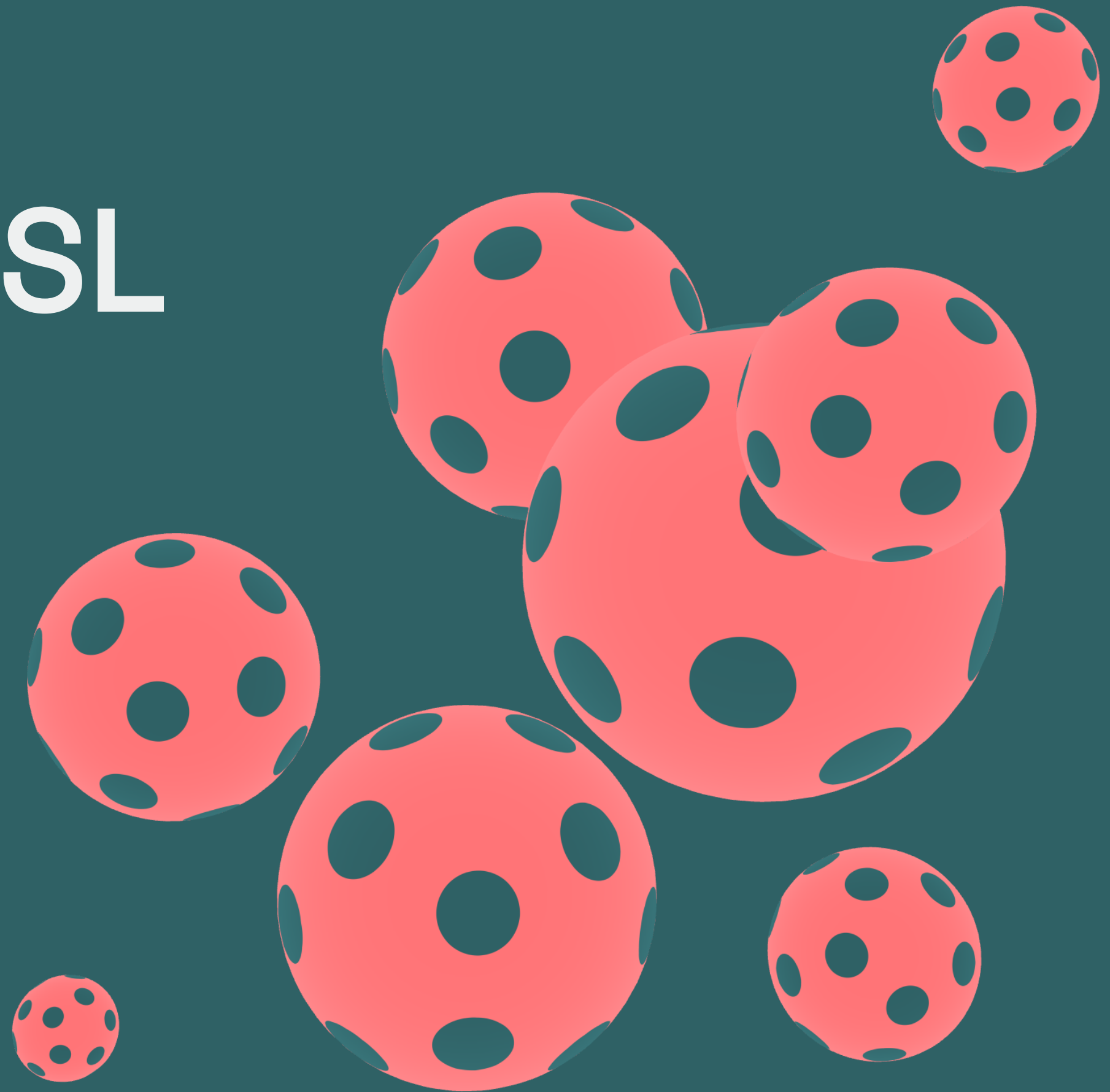


WebGL & GLSL

with Matt DesLauriers

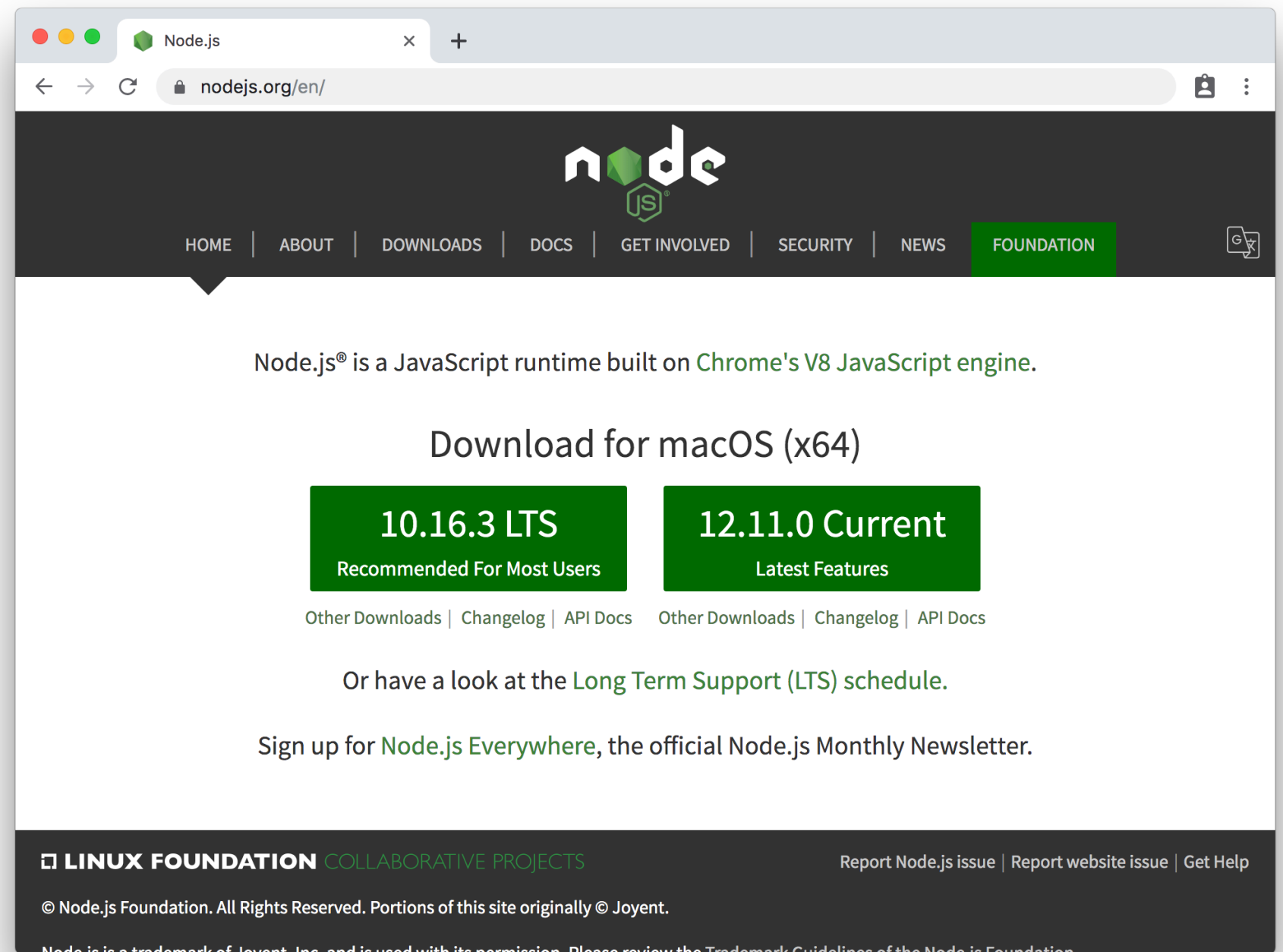


SCHEDULE

TOOLS

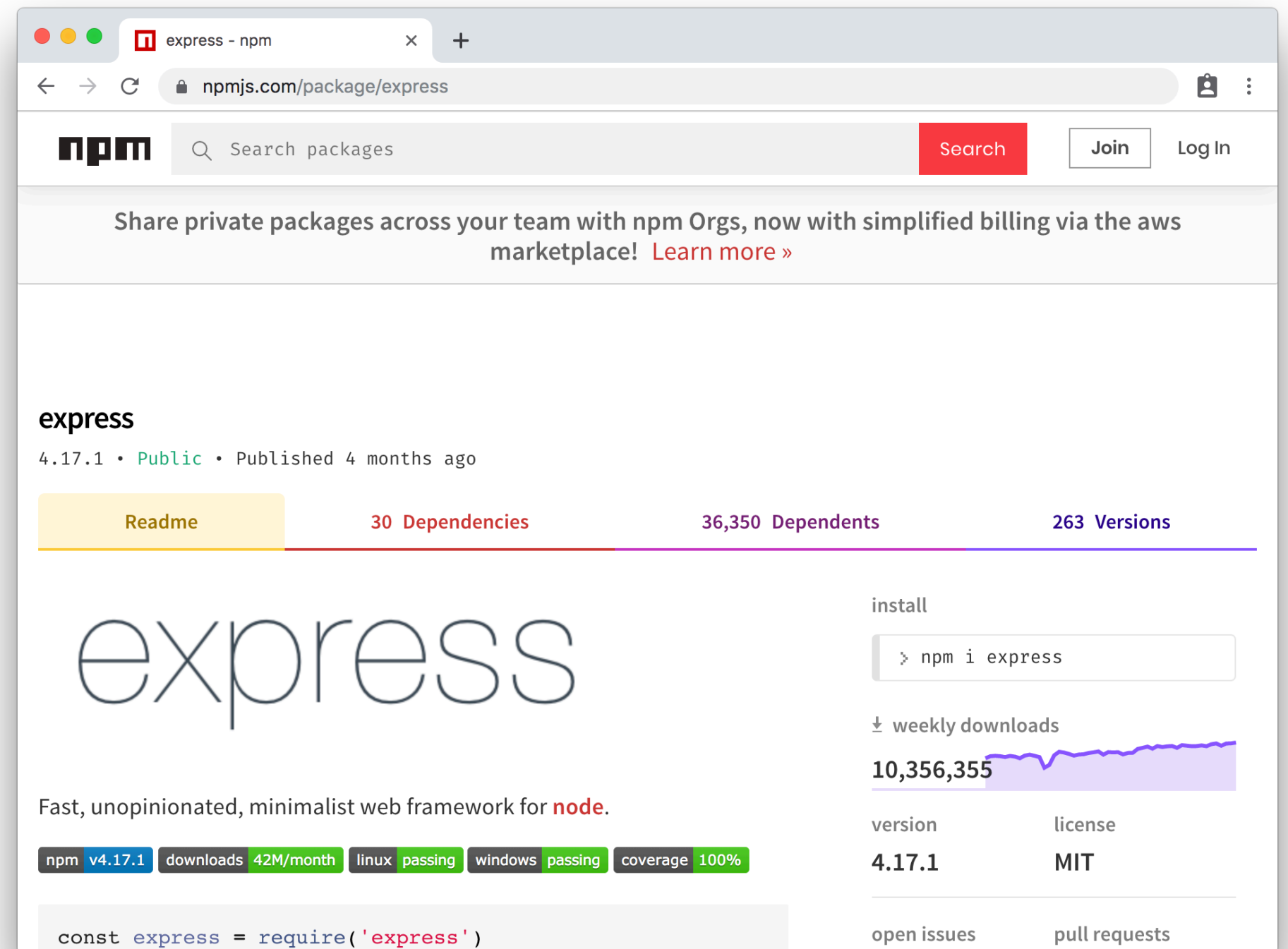
Node.js

A JavaScript runtime that allows us to build and run applications, work with files, and more.



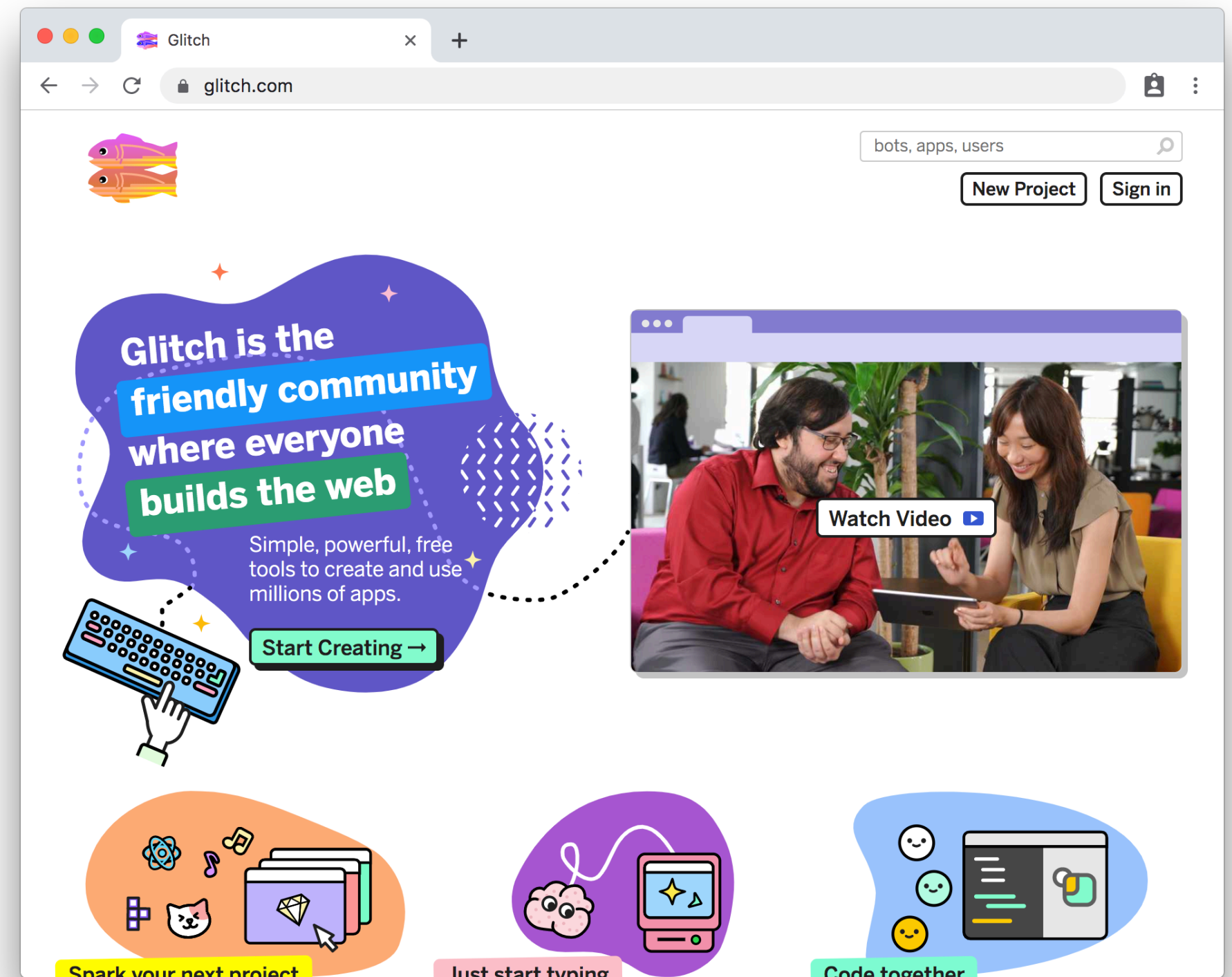
npm

An ecosystem that allows us to work with “modules” of code.



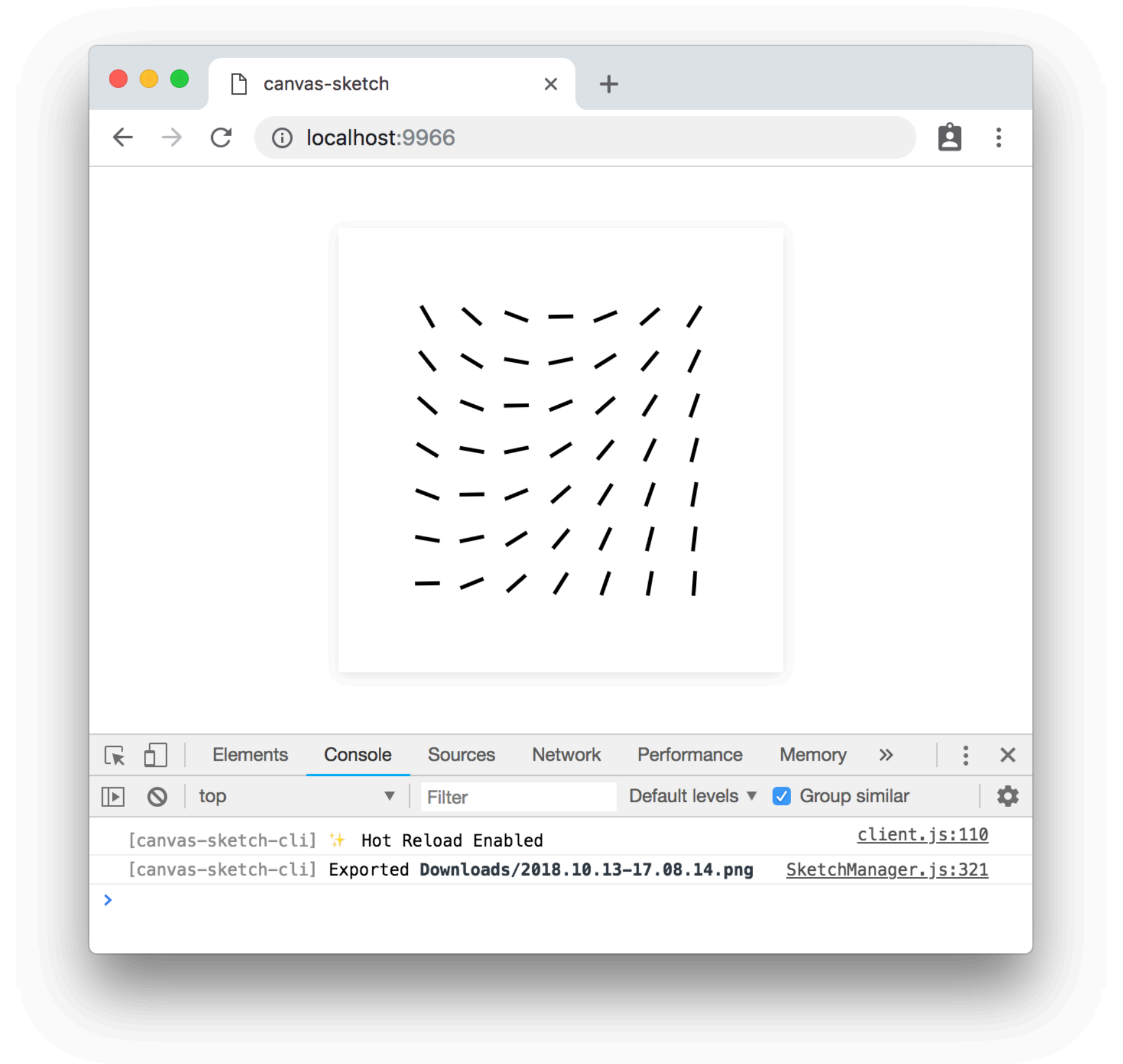
Glitch

A platform for building and sharing websites, demos, and creative projects in the browser.



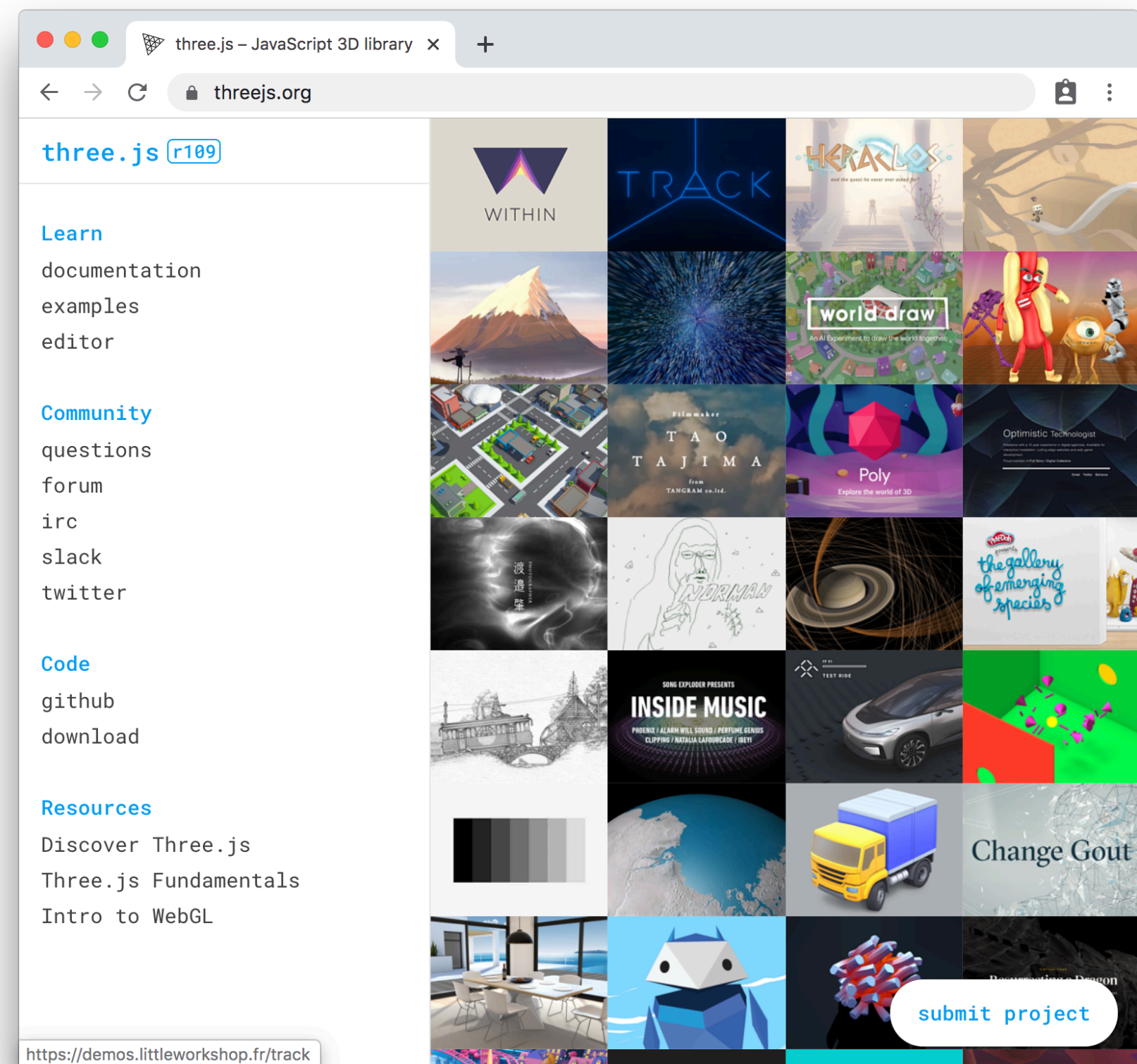
canvas-sketch

A framework for developing and exporting generative artwork for print, video, and web.



Three.js

A graphics engine for making 2D and 3D apps with JavaScript.



WEBGL

A browser implementation of OpenGL



The Industry's Foundation for High Performance Graphics

FROM GAMES TO VIRTUAL REALITY, MOBILE PHONES TO SUPERCOMPUTERS

Google Custom Search



[Documentation](#)

[Coding Resources](#)

[Wiki](#)

[Forums](#)

[About OpenGL](#)

OpenGL Headline News

[Submit News](#)

OpenGL Community Forums

The OpenGL Discussion Boards are now living within the [Khronos Community Forums](#). Everyone is encouraged to join in the conversation on OpenGL and the other Khronos Standards. If Slack is more your speed, we have a [Khronos Slack Group](#) you may join.

Oct 25, 2019 | [Read article...](#) | [Permalink](#)

Khronos Releases OpenVX 1.3

Today The Khronos Group announces the ratification and public release of the [OpenVX™ 1.3 specification](#), along with [code samples](#) and a prototype [conformance test suite](#). OpenVX is a royalty-free open standard for portable, optimized, and power-efficient vision and machine learning inferencing acceleration, vital to embedded and real-time use cases, such as face-, body-, and gesture-tracking, smart video surveillance, advanced driver assistance systems, object and scene reconstruction, augmented reality, visual inspection, robotics, and more. Also available today is an open sourced implementation of OpenVX 1.3 for Raspberry Pi to make OpenVX widely accessible to developers. The new specification can be found on the [OpenVX registry](#). Read the [press release](#) for more details and give Khronos feedback on the [OpenVX community forums](#).

Oct 25, 2019 | [Read article...](#) | [Permalink](#)

NeoAxis Engine 2019.1 Announced



Download OpenGL

[Getting Started with OpenGL](#)

[Official OpenGL 4.6 feedback thread](#)

[OpenGL Reference Cards](#)

[OpenGL Registry](#)

[OpenGL Conformant Products](#)




[Getting Started with Vulkan](#)

[Vulkan Reference Cards](#)



[Getting Started with OpenGL ES](#)



DOOM (2016)

GLSL

A shader language for OpenGL



```
uniform vec3 color;  
uniform float opacity;
```

```
varying vec2 vUv;
```

```
void main () {  
    vec3 fragColor = vec3(vUv.x) * color;  
    gl_FragColor = vec4(fragColor, opacity);  
}
```



```
const fragmentShader = `  
    uniform vec3 color;  
    uniform float opacity;  
  
    varying vec2 vUv;  
  
    void main () {  
        vec3 fragColor = vec3(vUv.x) * color;  
        gl_FragColor = vec4(fragColor, opacity);  
    }  
`;  
;
```



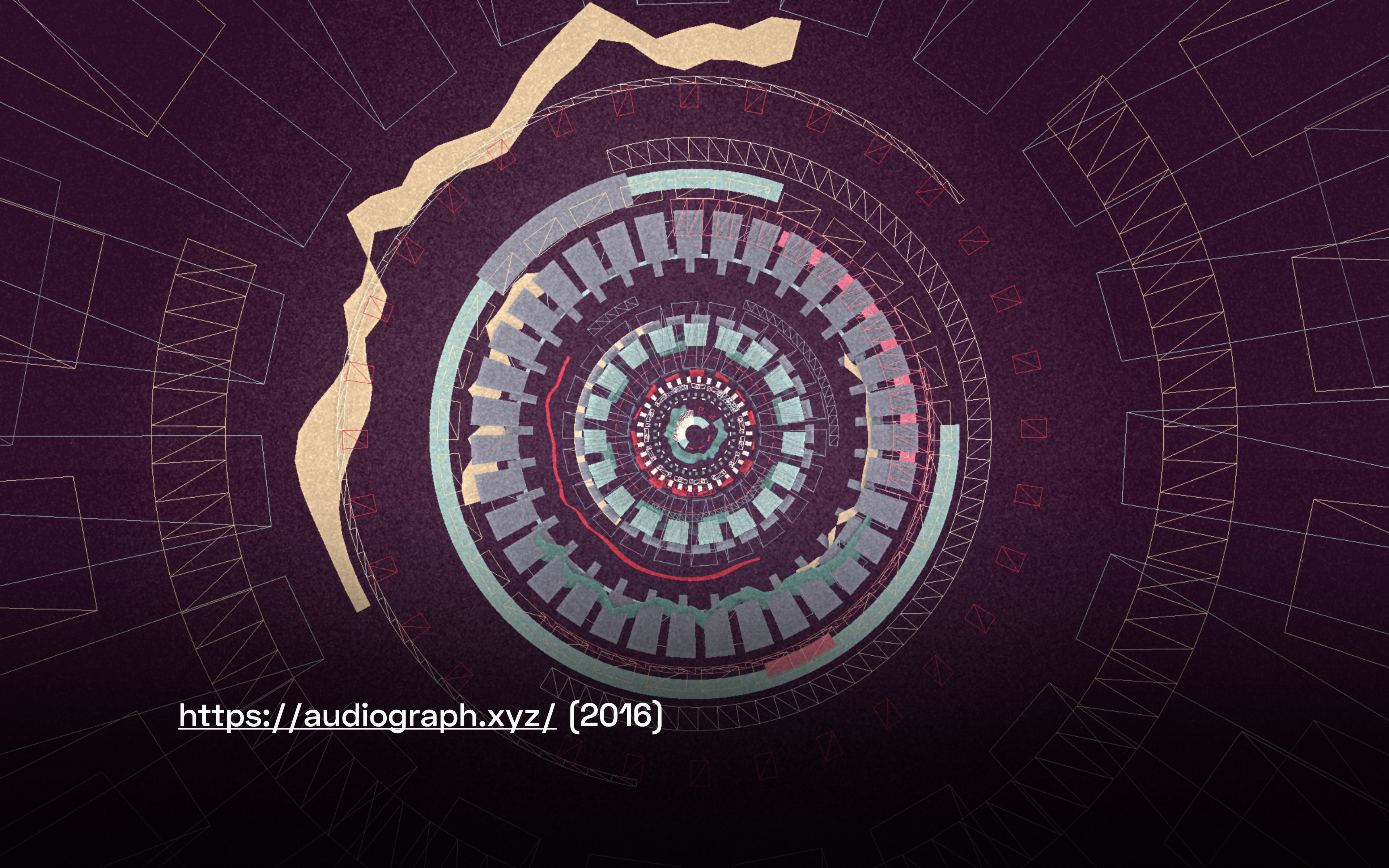
Three.js

A high-level framework atop WebGL/GLSL

A decorative border composed of short, white, slightly curved line segments arranged in a circular pattern around the central text.

EXAMPLES

First, some of my own work



<https://audiograph.xyz/> (2016)

A surreal landscape featuring several large, dark, jagged rock islands floating in a vast, hazy sky. The sky is a mix of soft pinks, oranges, and blues, suggesting a sunset or sunrise. A large flock of small, dark birds is flying in a curved path across the upper left portion of the sky. The rock islands are silhouetted against the bright background, with some small trees visible on the top of the central island. The overall mood is dreamlike and ethereal.

In Limbo (2016)

Jam3 × Mozilla × Matt DesLauriers



Art, Story & Experience

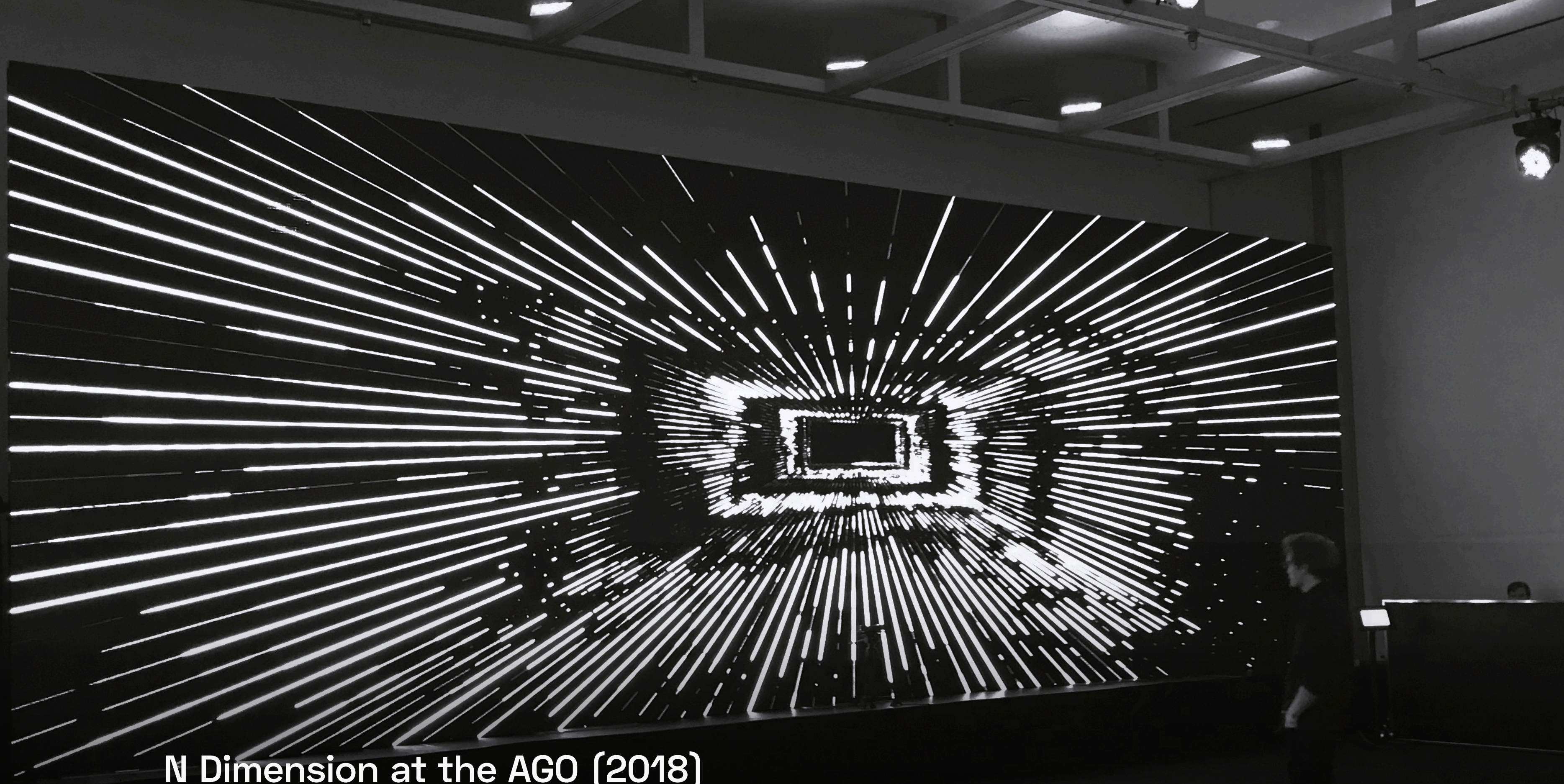
We bring all visual mediums to life.

Leaf Notes (2018)

Tendrill × Matt DesLauriers



KIKK AR Exhibition in Namur, Belgium (2018)



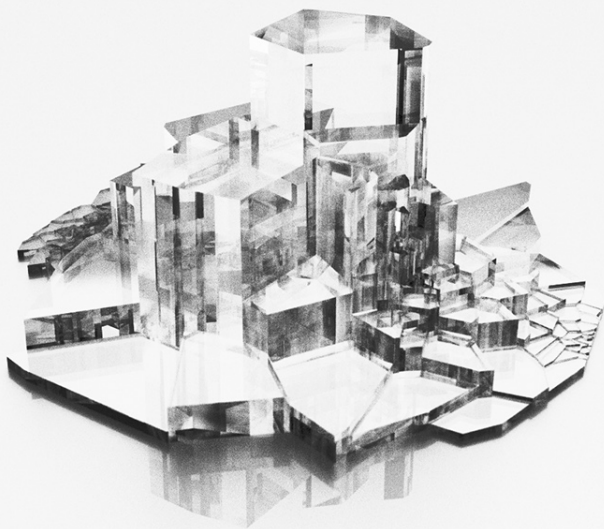
N Dimension at the AGO (2018)

Array of Stars × Matt DesLauriers

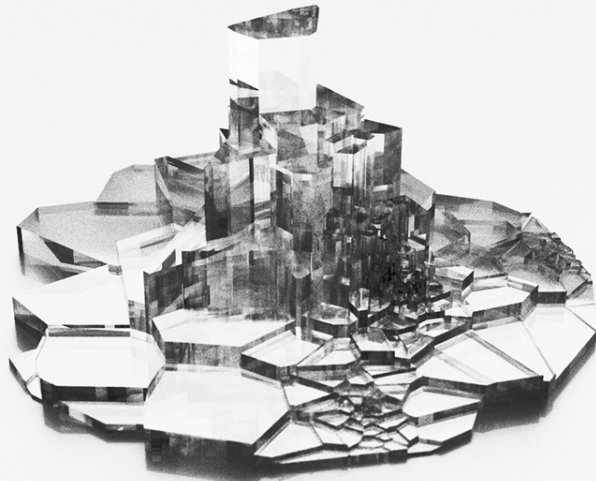




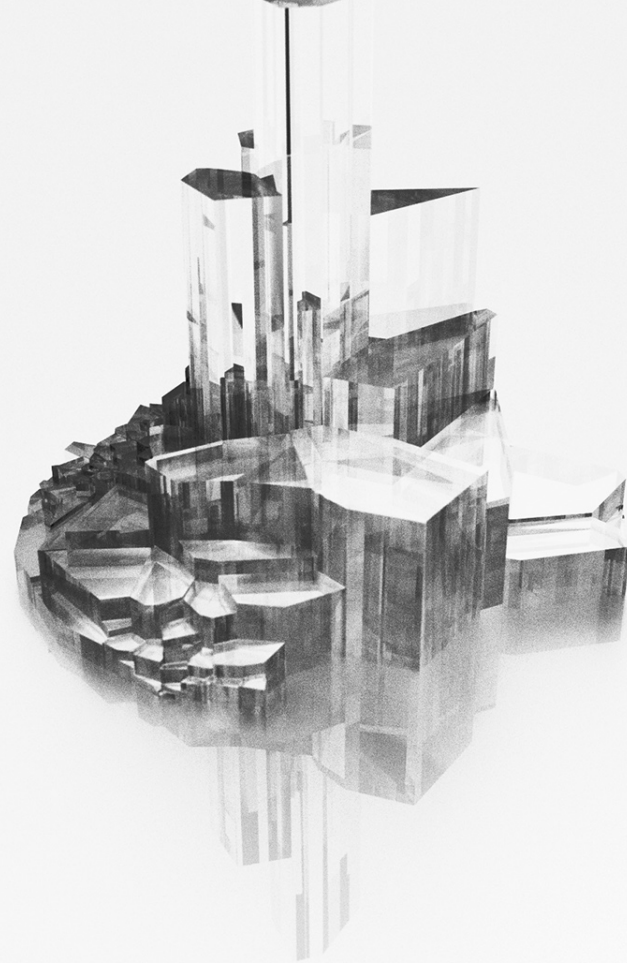
NEW YORK CITY



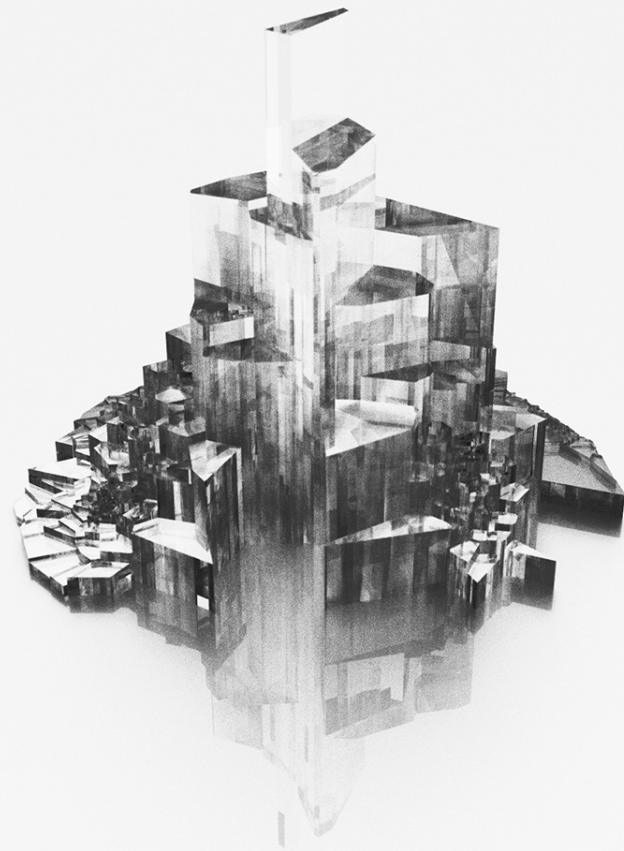
LONDON



SYDNEY



SHANGHAI



HONG KONG

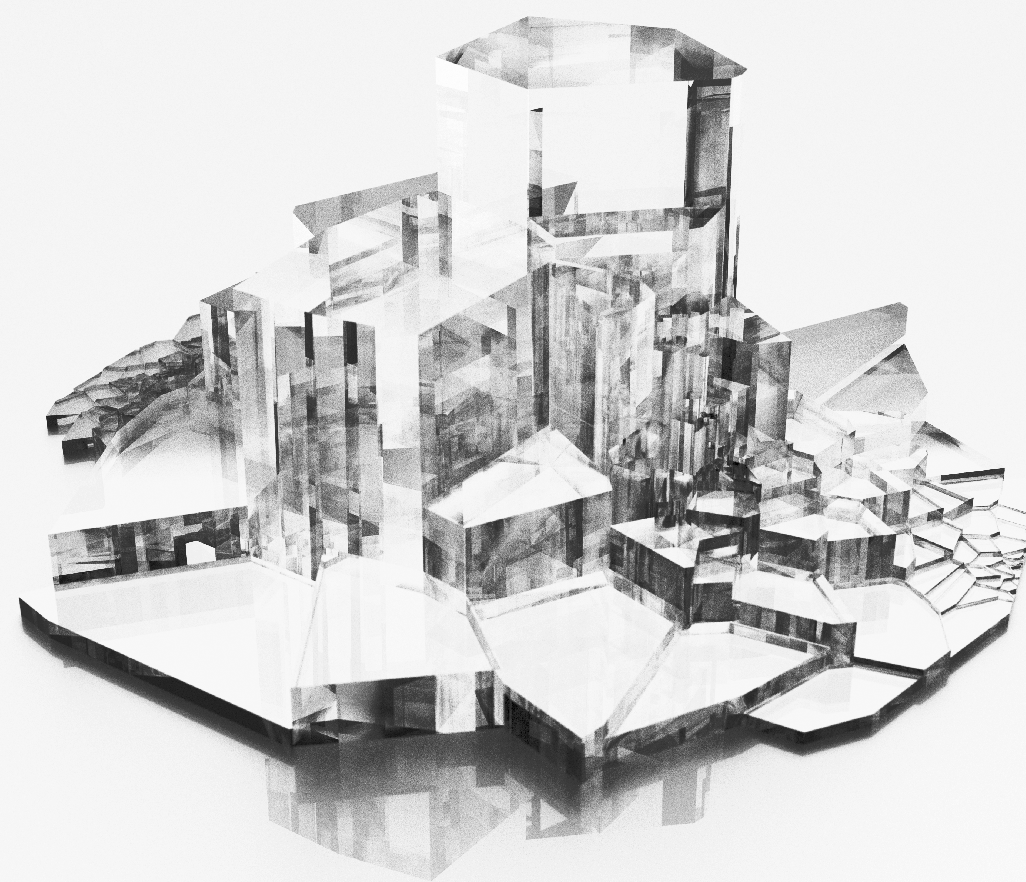


Crystal Towers (2017)
Matt DesLauriers

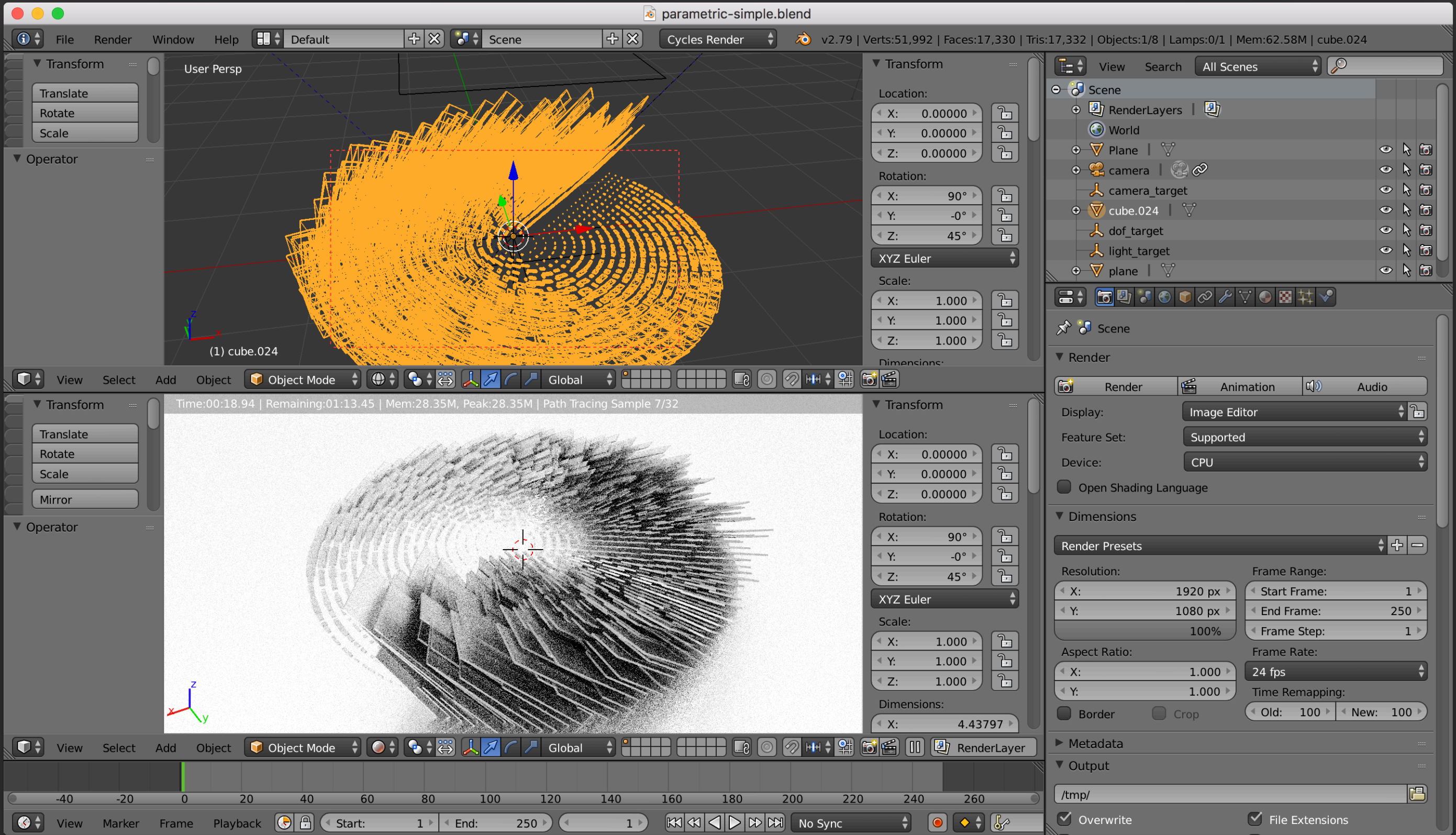




NEW YORK CITY



LONDON

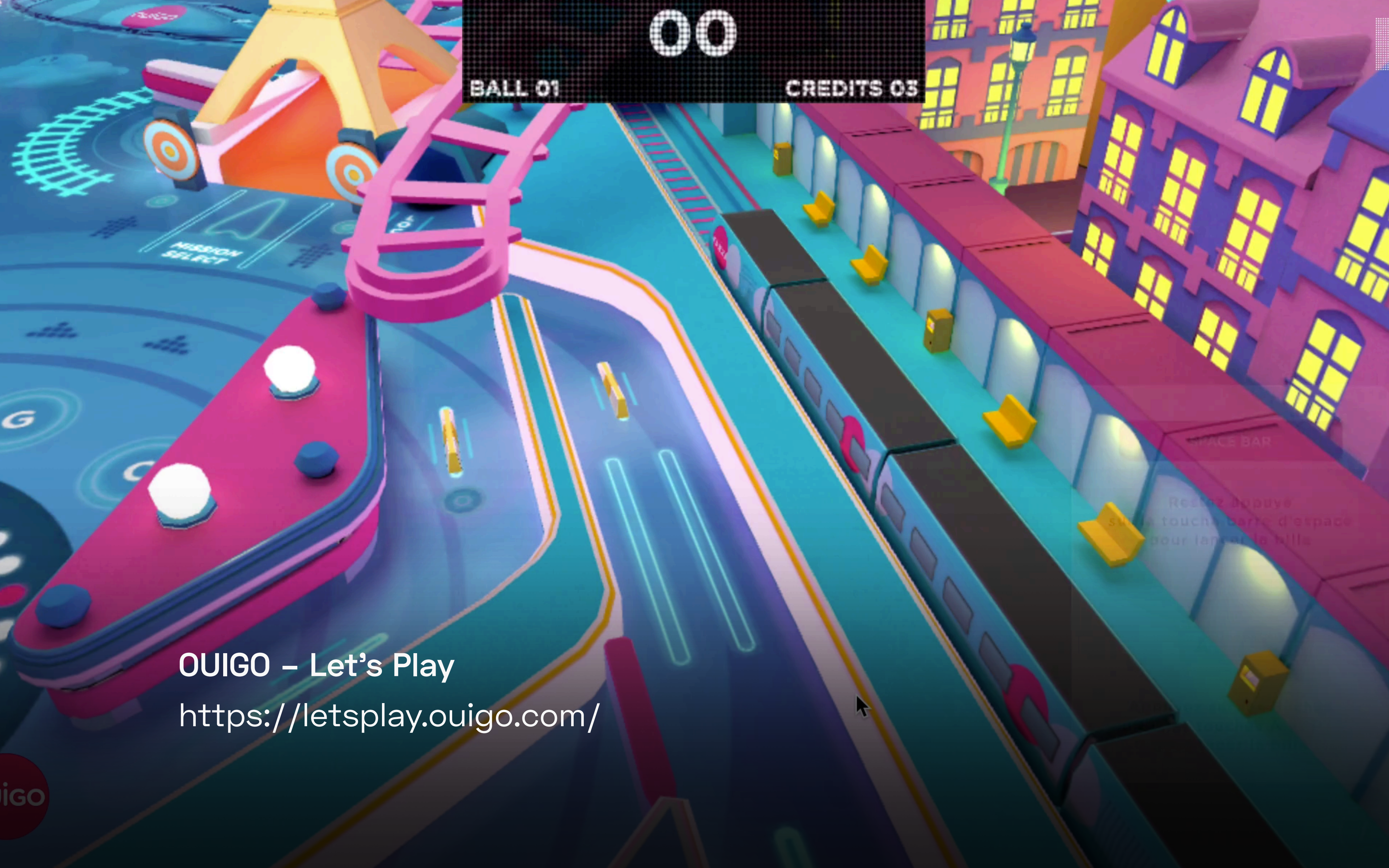






EXAMPLES

Cool WebGL projects in the wild



00

BALL 01

CREDITS 03

OUIGO – Let's Play

<https://letsplay.ouigo.com/>

SPACE BAR

Revenez appuyez
sur la touche barre d'espace
pour lancer le bille

ouigo

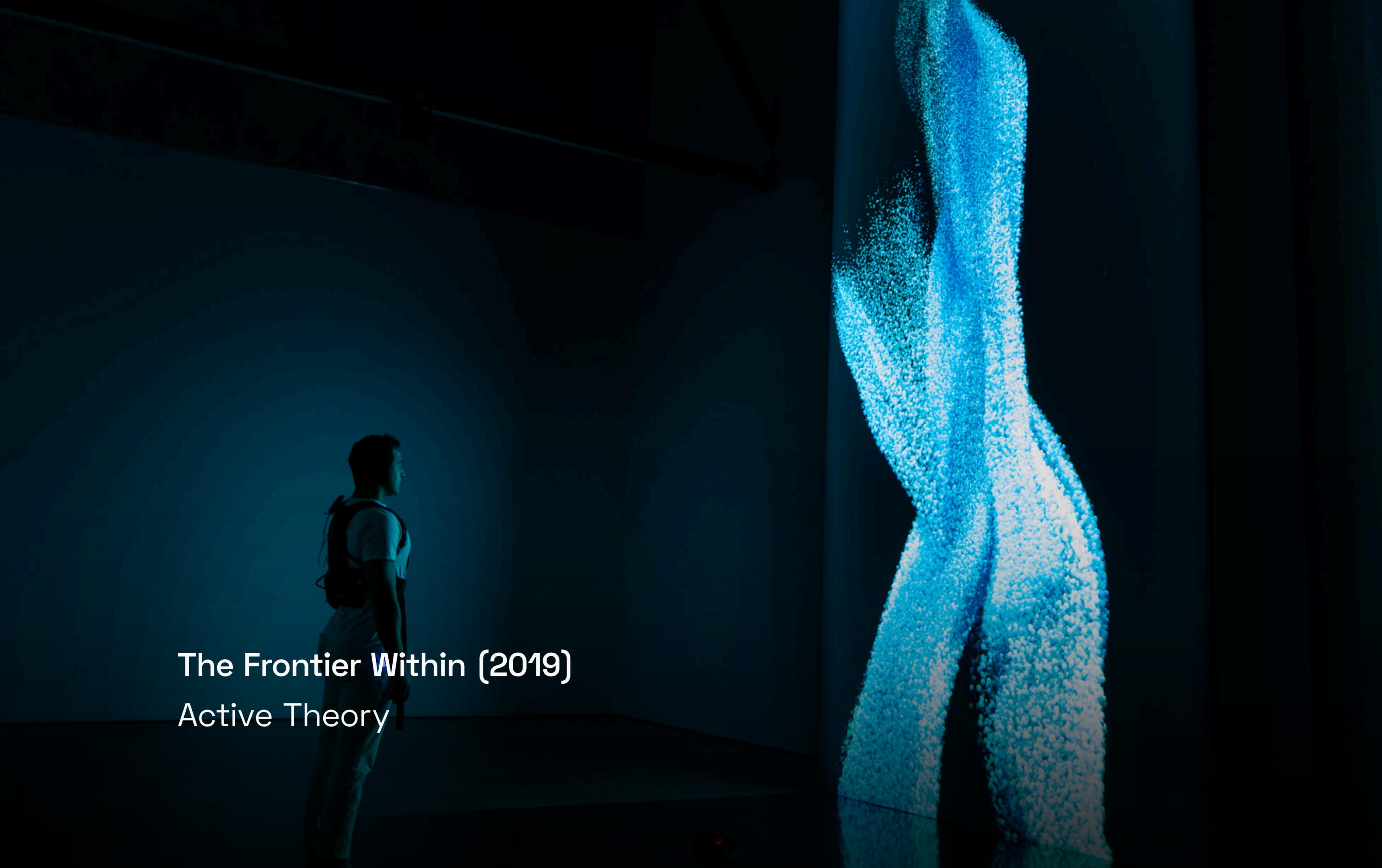


Bruno Simon's Portfolio
<https://bruno-simon.com/>



Hertzian Landscapes (2019)

Richard Vijgen



The Frontier Within (2019)
Active Theory



NOT WEBGL

But they could have been

Future You
Universal Everything





Constellations (2018)
Joanie Lemercier

A dark, futuristic interior scene. Two silhouetted figures stand on either side of a small table. The ceiling and walls are composed of a complex, glowing blue light structure that creates a series of sharp, intersecting lines and bright points of light. The overall atmosphere is mysterious and high-tech.

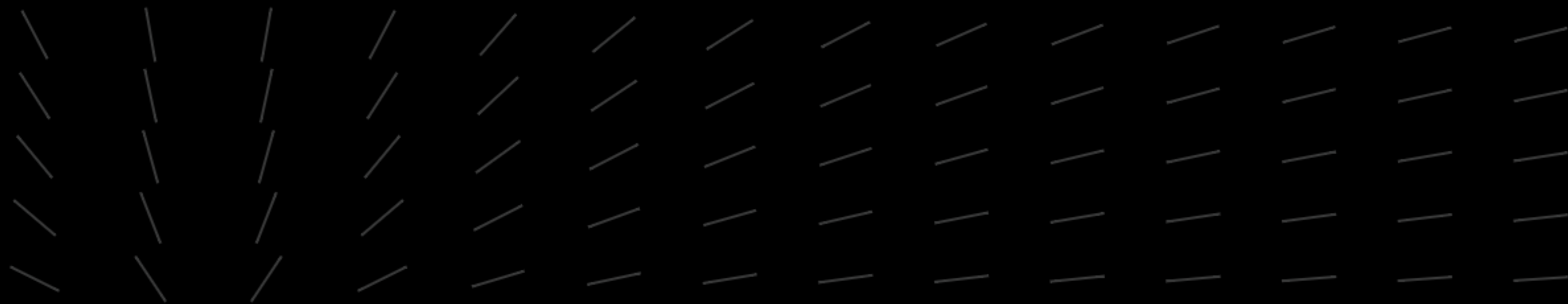
NONOTAK

<https://vimeo.com/79732146>

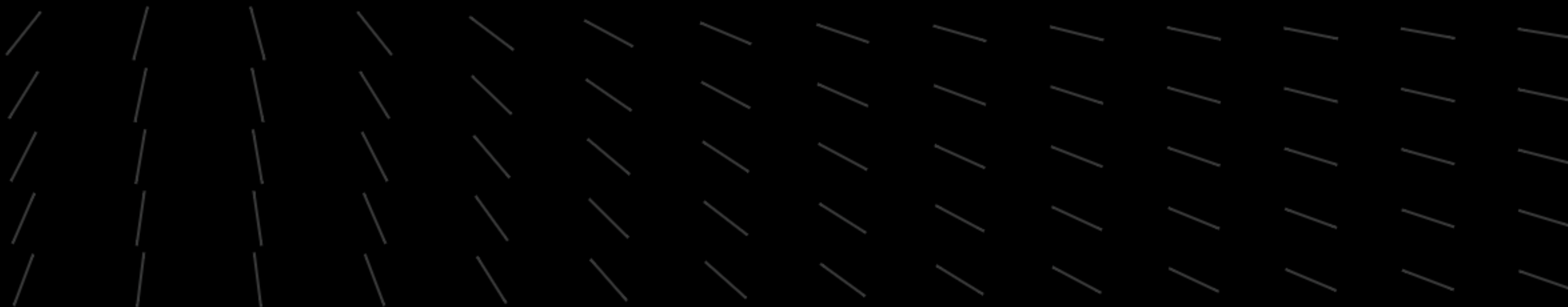


TIME TO DIVE IN..

Let's see the course repository



<https://github.com/mattdesl/workshop-webgl-glsl>



A decorative graphic consisting of numerous short, white line segments arranged in a circular pattern around the central text. The lines are of varying lengths and orientations, creating a sense of motion or a stylized circular border.

GUIDE BOOK

Getting started with 3D

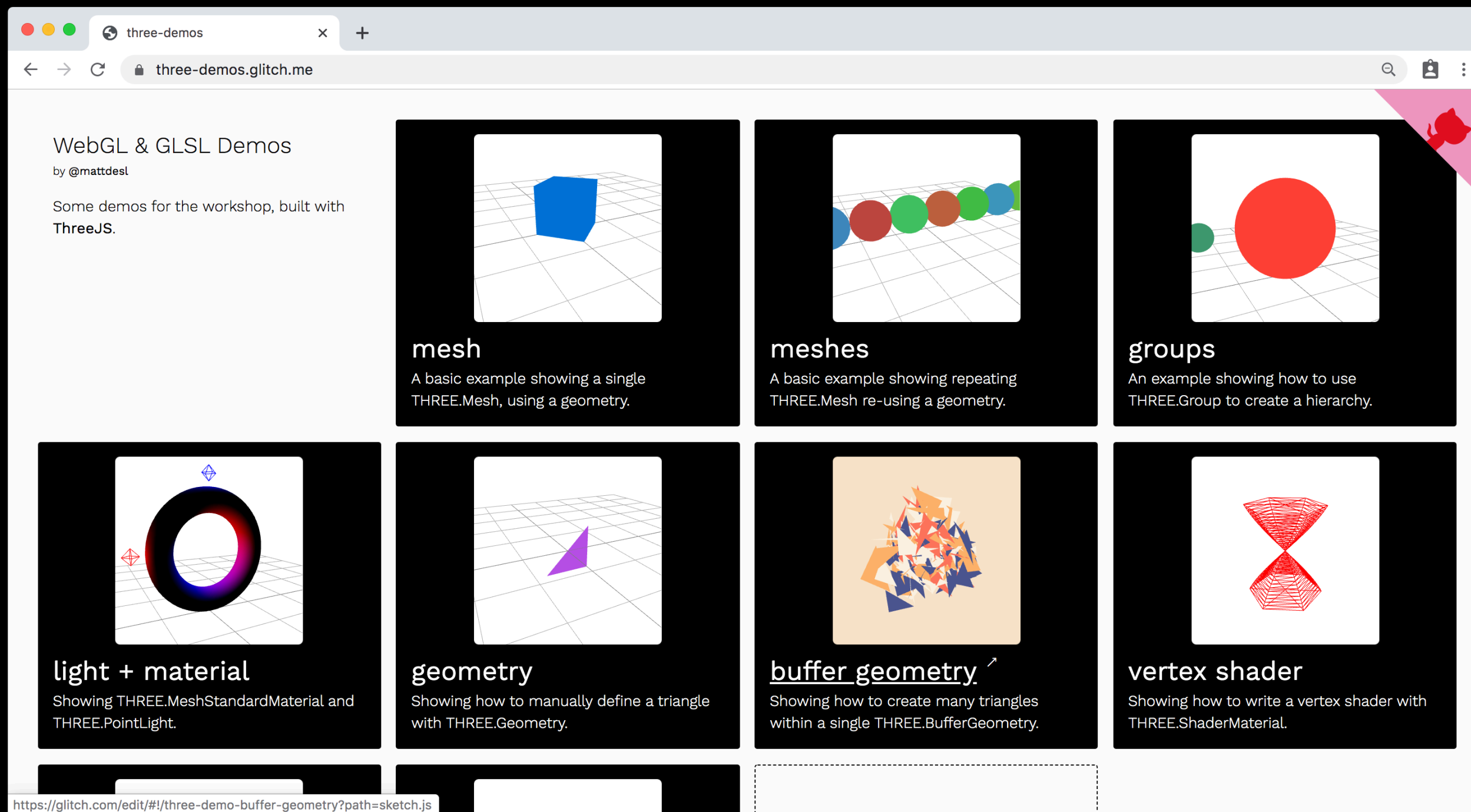


SETUP

Node.js, npm, and more!



LET'S CODE!



<https://three-demos.glitch.me>





LUNCH !

SHADERS

The background features a series of short, white, diagonal line segments arranged in a circular pattern around the central text. These segments are oriented at various angles, creating a sense of motion or a stylized circular border.

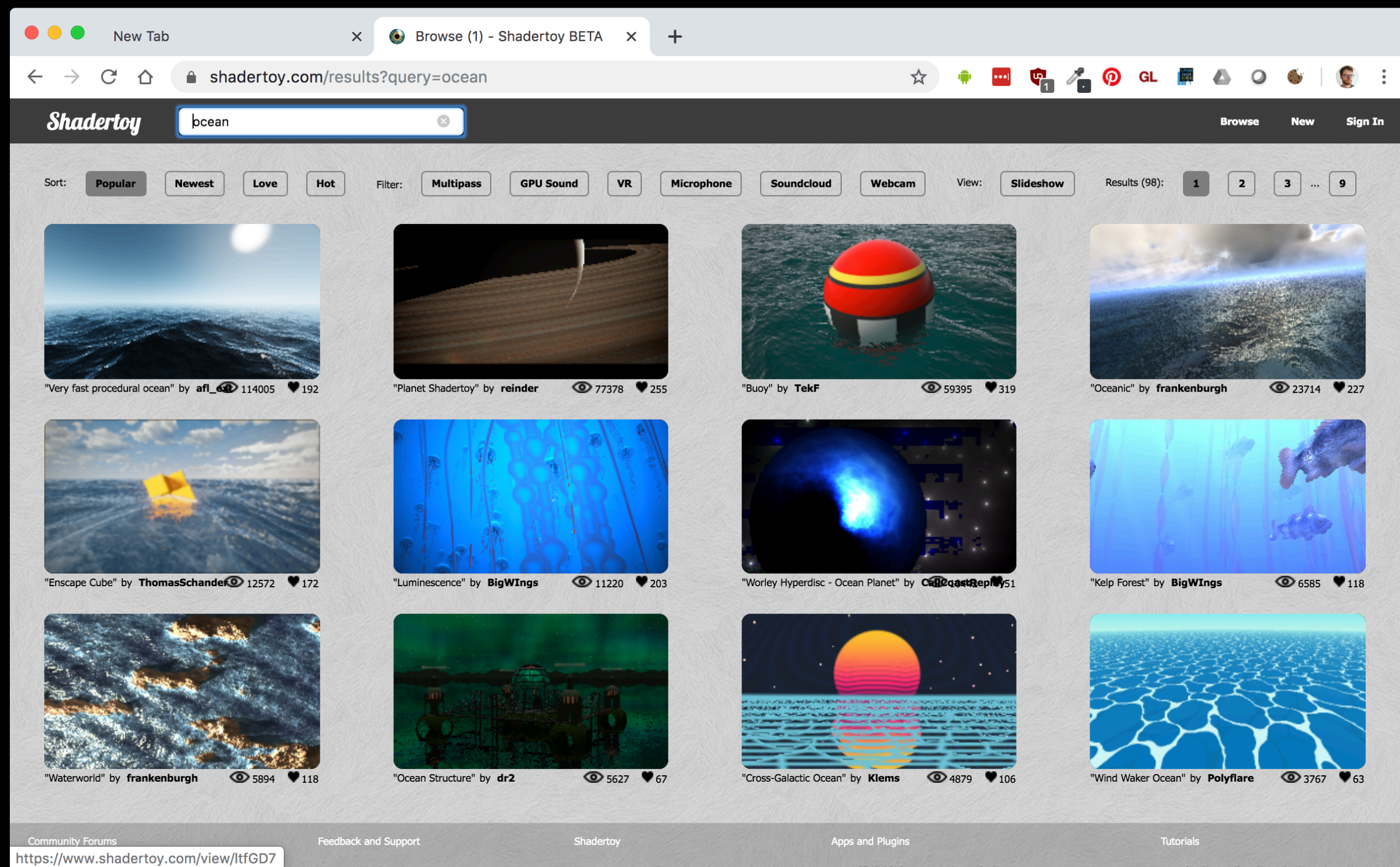
TINY PROGRAMS

Run in parallel on the GPU



```
uniform float opacity;  
uniform vec3 color;
```

```
void main () {  
    gl_FragColor = vec4(color, opacity);  
}
```



<https://shadertoy.com/>



GUIDE BOOK

Getting started with shaders



SNIPPETS

If your'e lost...

three-demos

sketch.js – three-demo-vertex- x

+

← → ↺ ⌂

glitch.com/edit/#!/three-demo-vertex-shader?path=sketch.js:1:0

🔍 ☆ 🤖 ⋮ 🔑 📌 GL 📺 🔍 ⌂ 🍪 👤 ⋮

three-demo-vertex-shad... Show sketch.js

👤 Share ↕

New File ▾

assets

styles/

.eslintrc.json

embed.html

index.html

sketch.js

Format This File ✨

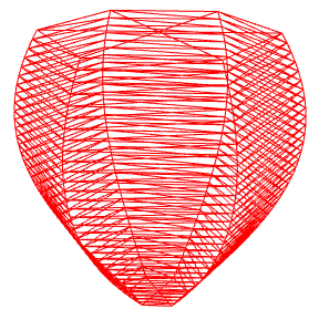
```
1~ const settings = {
2  // Make the loop animated unless ?static is passed to URL
3  animate: !/static/i.test(window.location.search),
4  // Get a WebGL canvas rather than 2D
5  context: "webgl"
6 };
7
8~ const sketch = ({ context }) => {
9  // Create a renderer
10~ const renderer = new THREE.WebGLRenderer({
11    context
12  });
13
14  // WebGL background color
15  renderer.setClearColor("#fff", 1);
16
17  // Setup a camera
18  const camera = new THREE.PerspectiveCamera(45, 1, 0.01, 100);
19  camera.position.set(2, 2, -4);
20  camera.lookAt(new THREE.Vector3());
21
22  // Setup camera controller
23  const controls = new THREE.OrbitControls(camera, context.ca
24  controls.enableZoom = false;
25
26  // Setup your scene
27  const scene = new THREE.Scene();
28
29  // We can apply vertex shaders to built-in geometries to ma
30  // them a bit more interesting
31  const geometry = new THREE.CylinderGeometry(1, 1, 2, 6, 32)
32
33  // Here's how we define a shader material
```

Change URL ↺

×

vertex shader

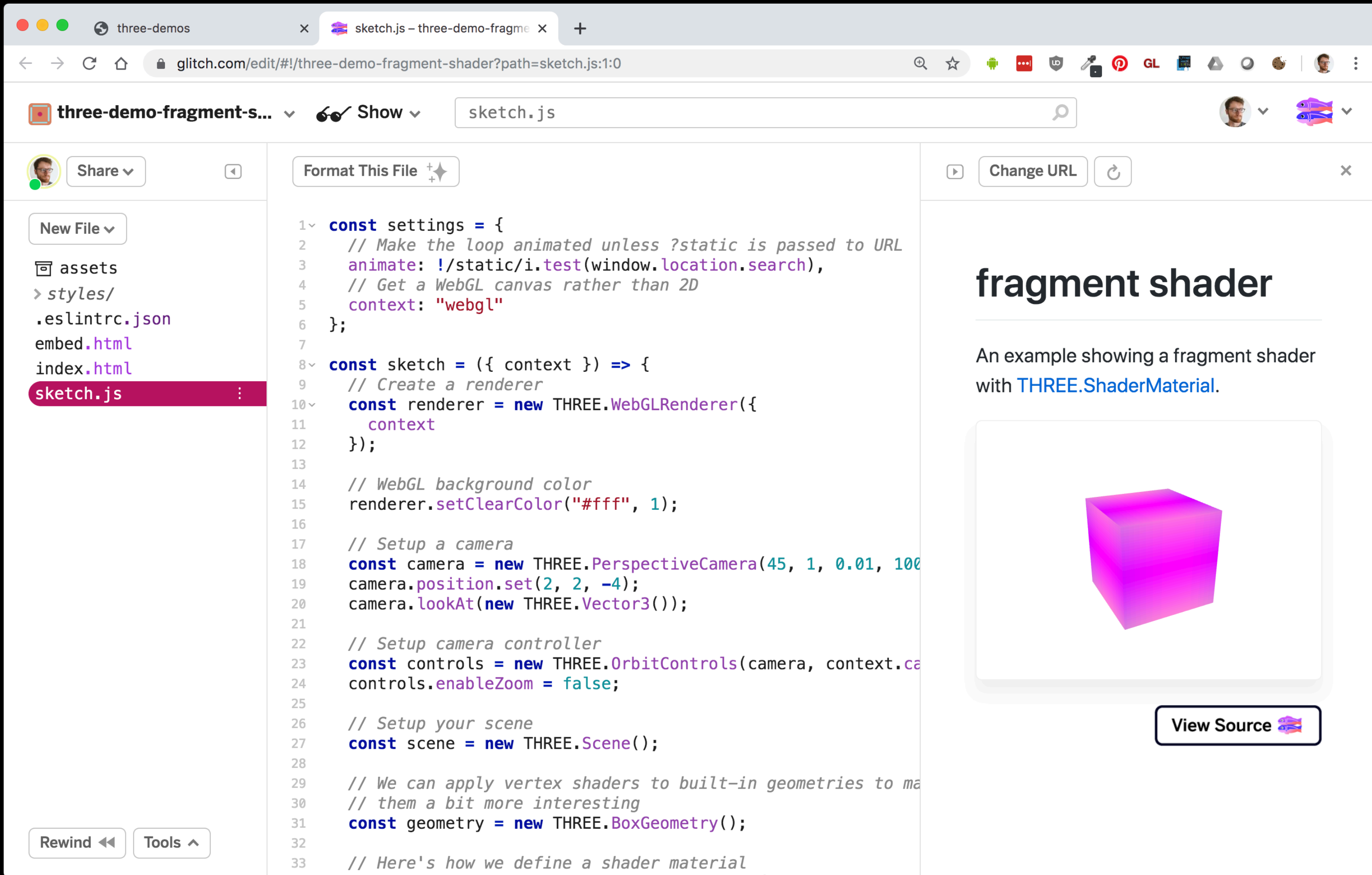
An example showing a vertex shader with [THREE.ShaderMaterial](#).



View Source 🐟

Rewind ⏮

Tools ⌆

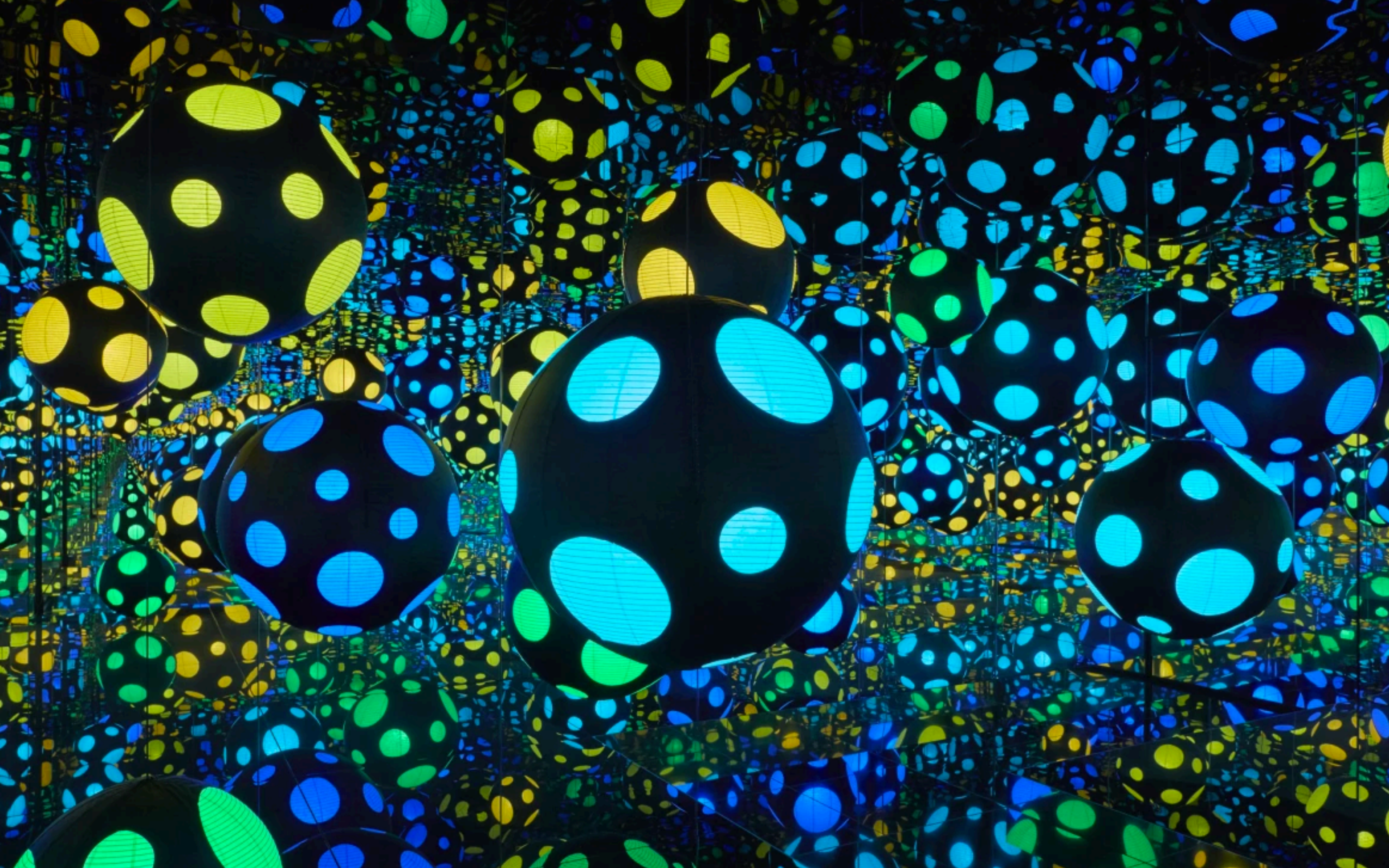




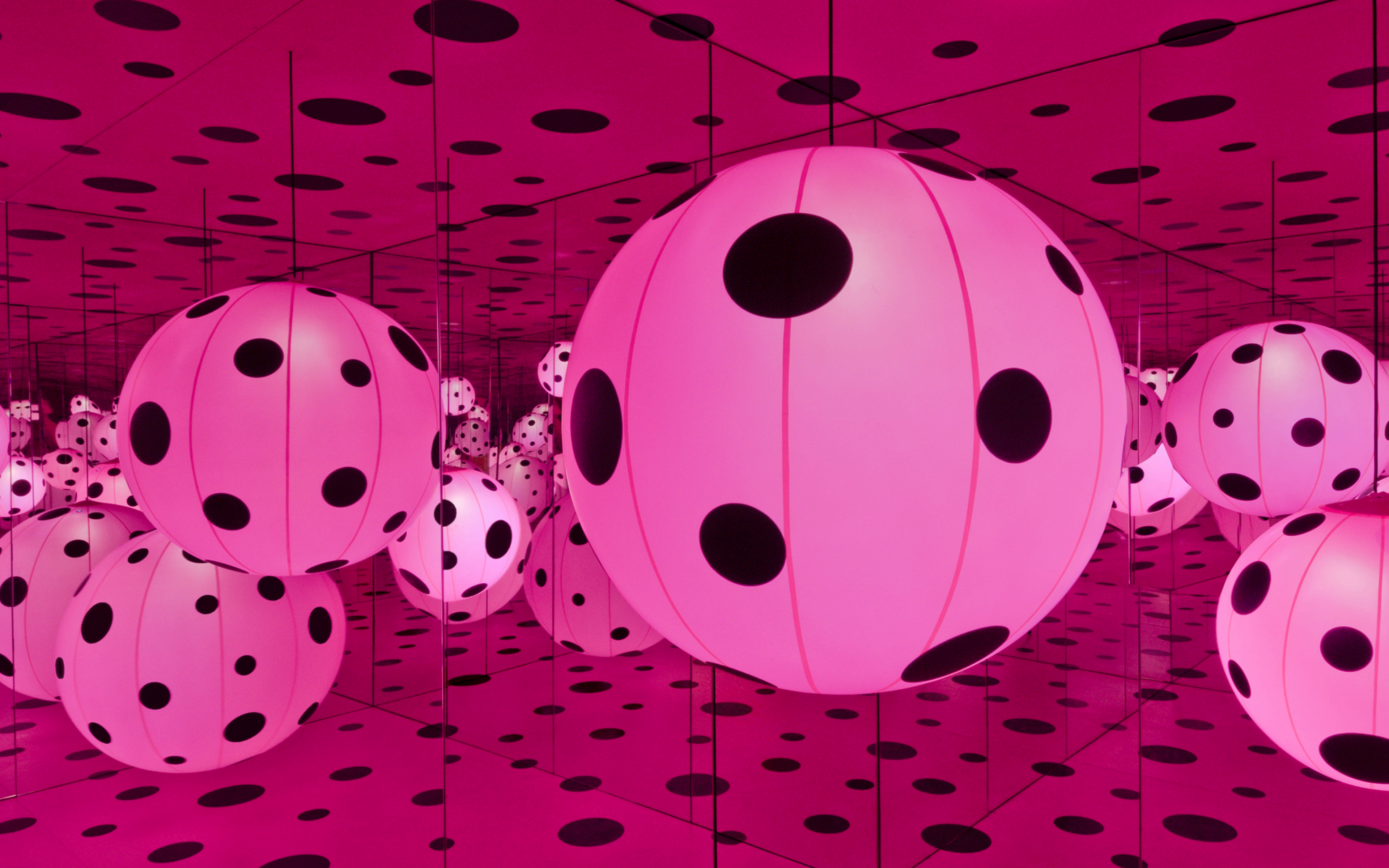
FINDING INSPIRATION



Yayoi Kusama (b. 1929)

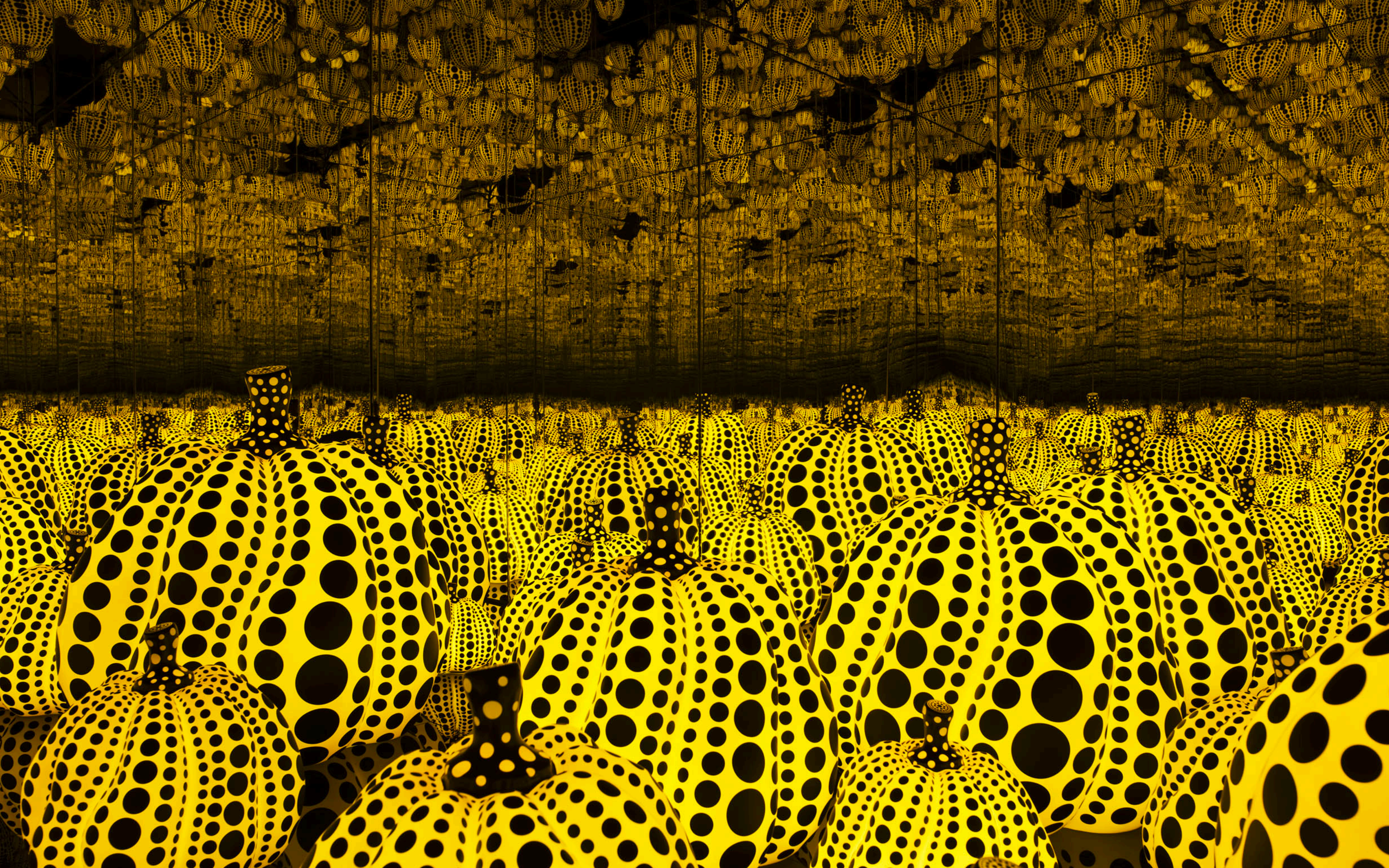




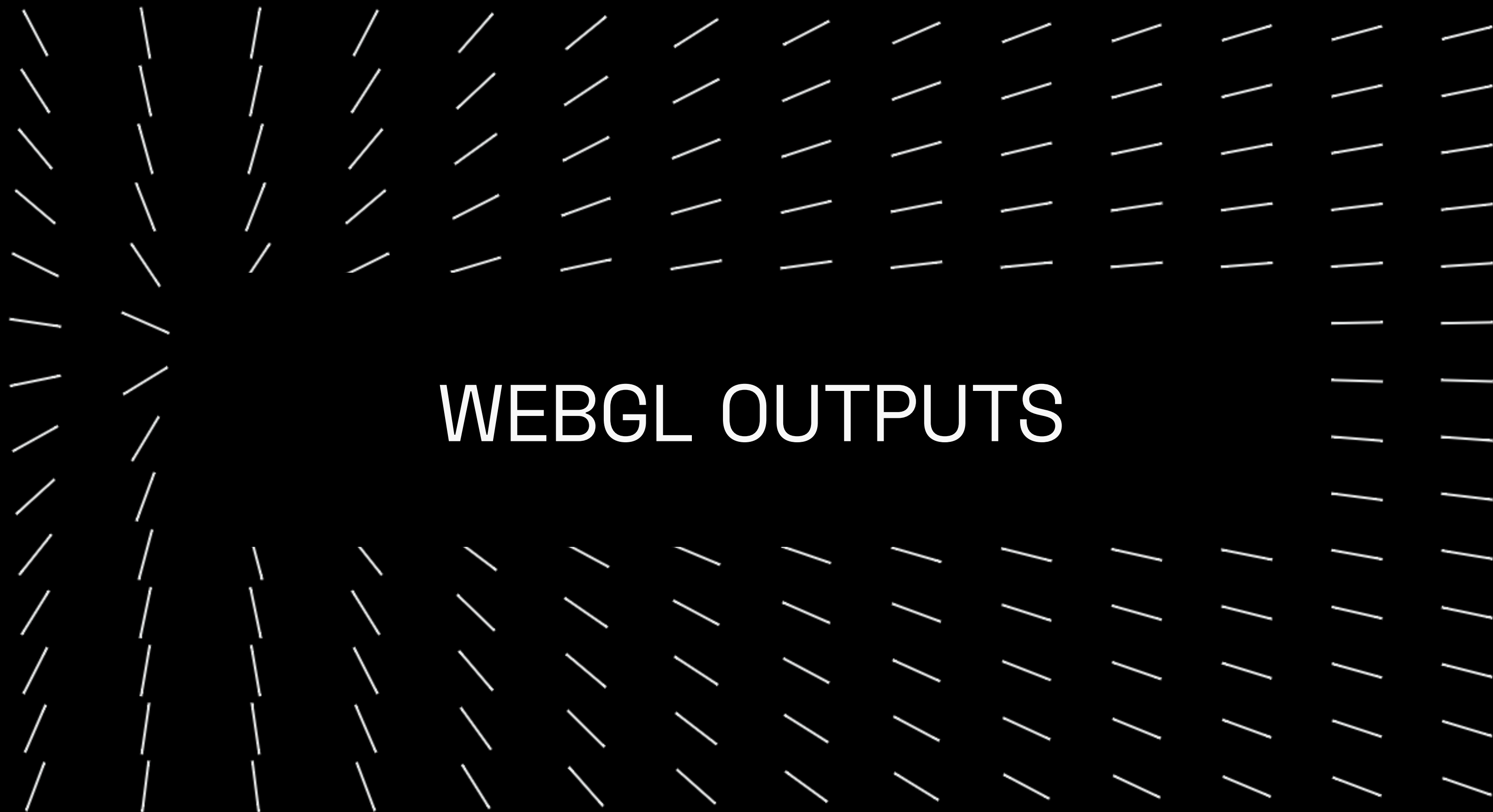


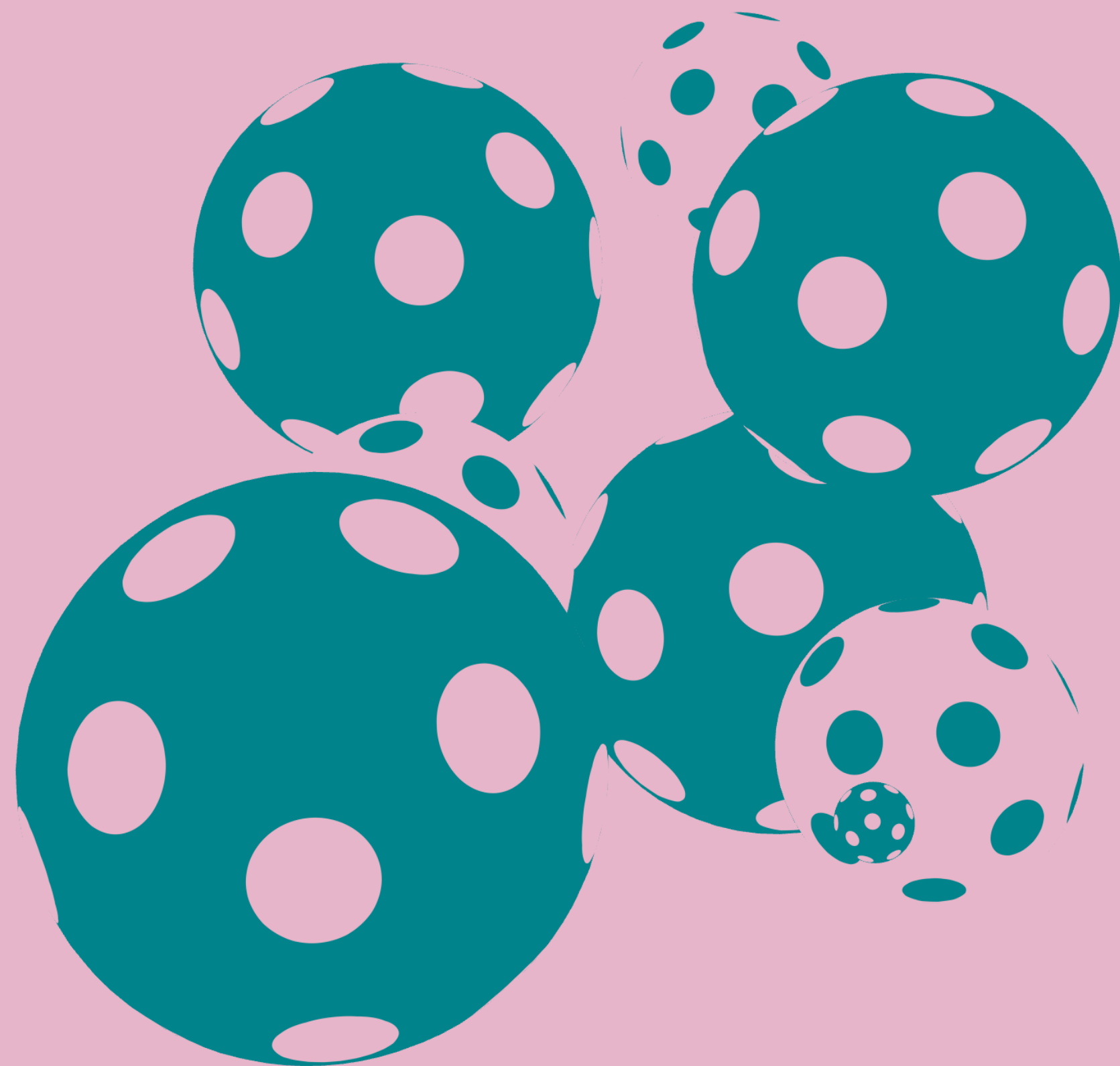


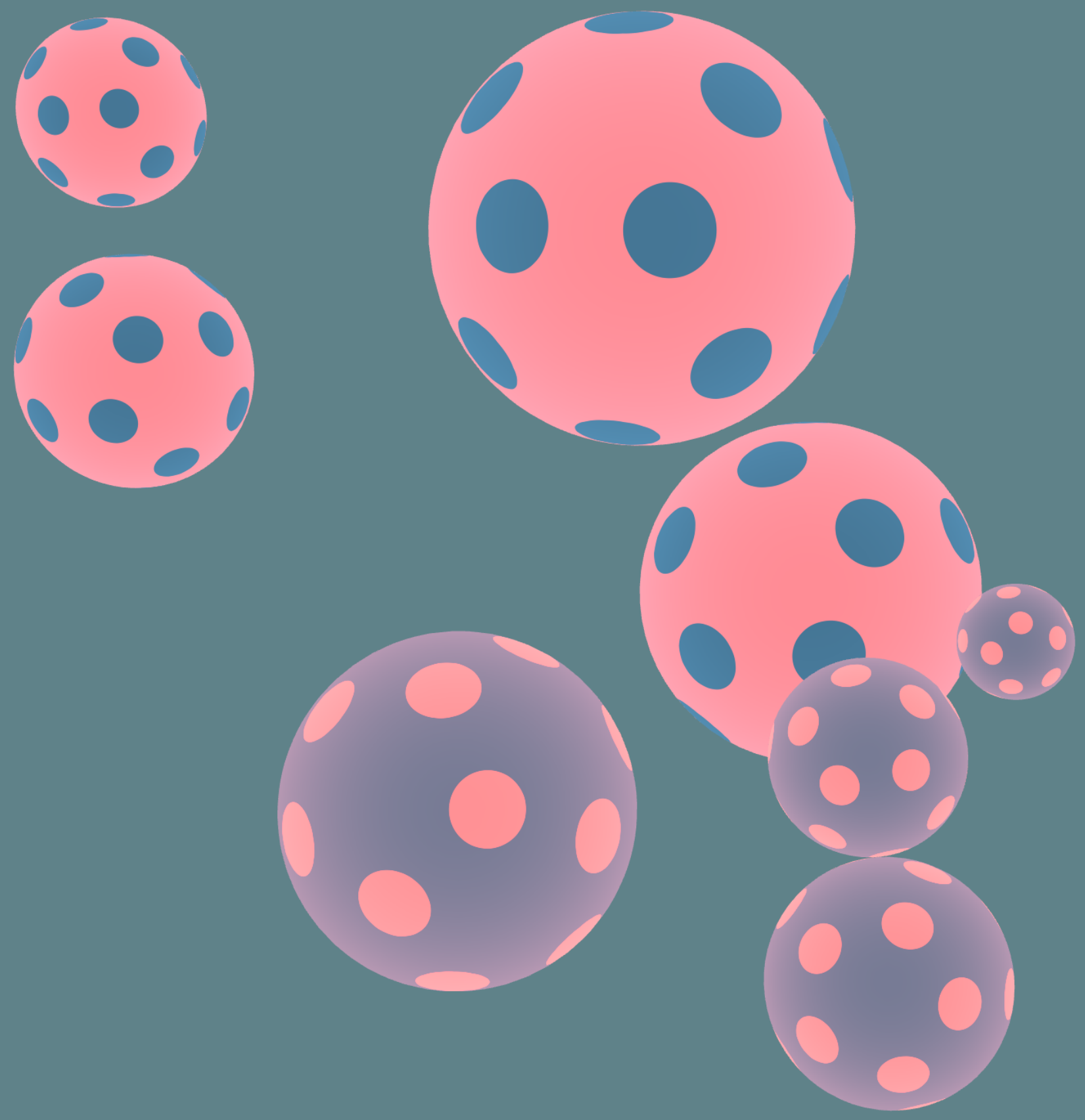


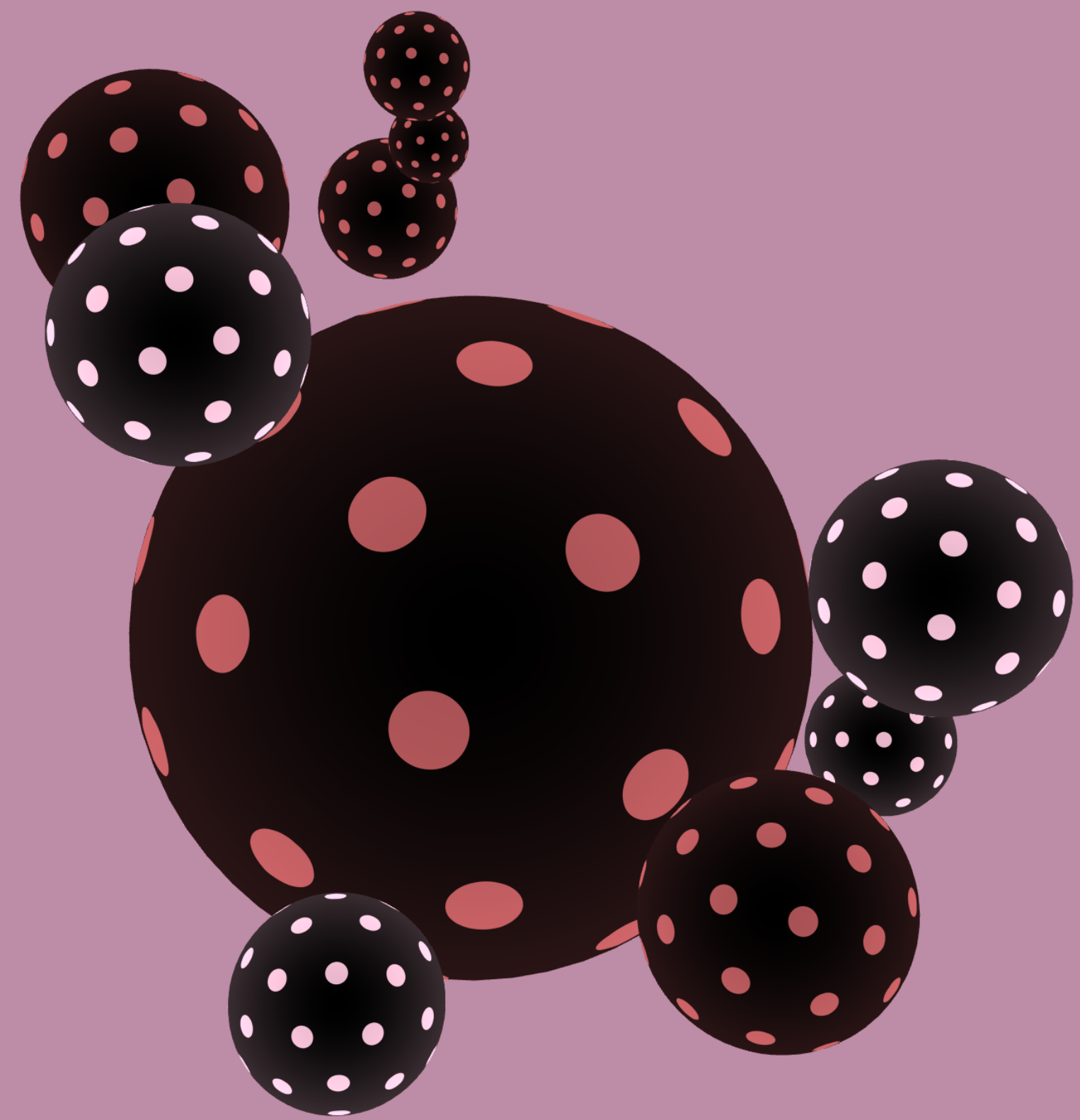


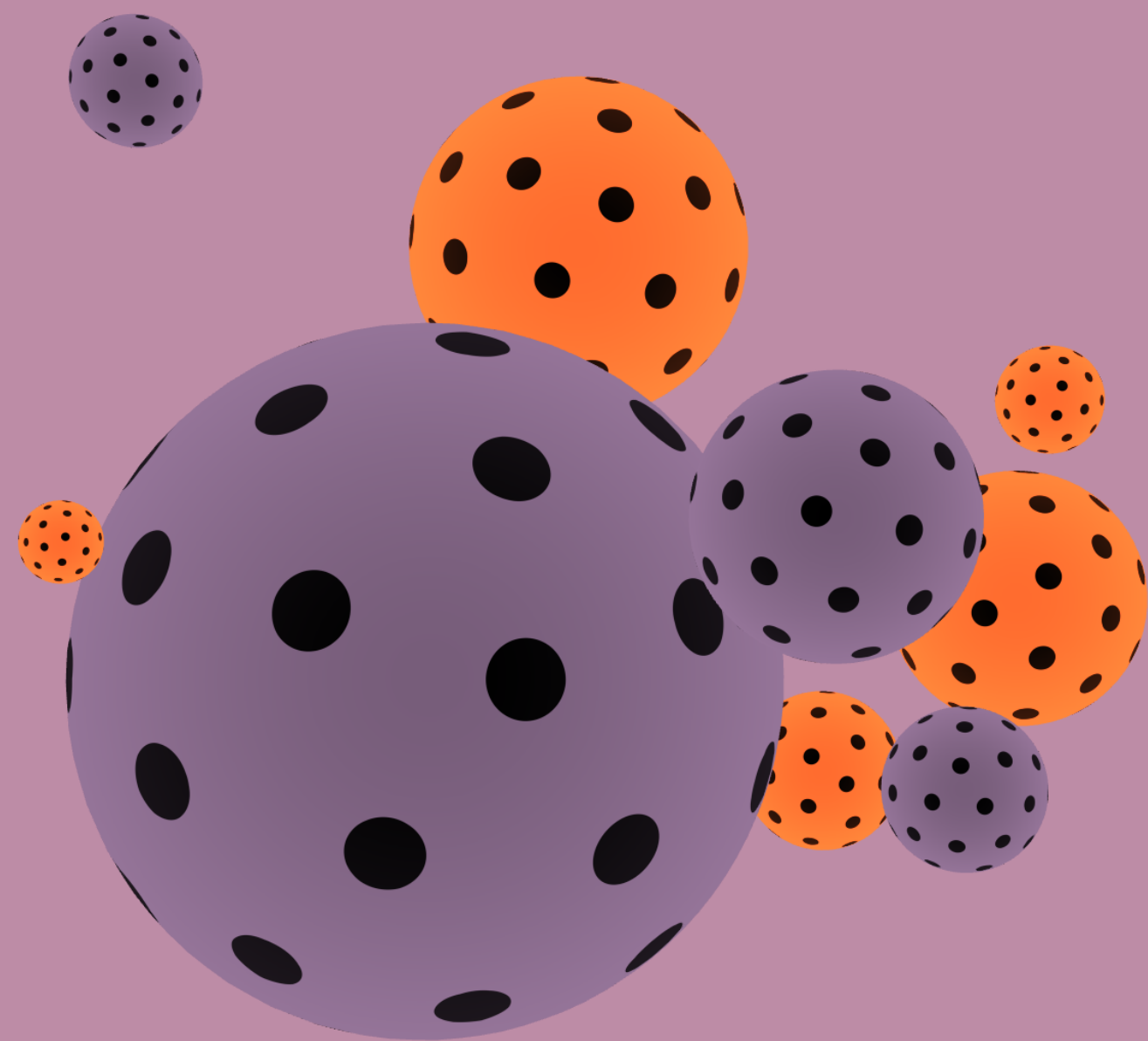
WEBGL OUTPUTS

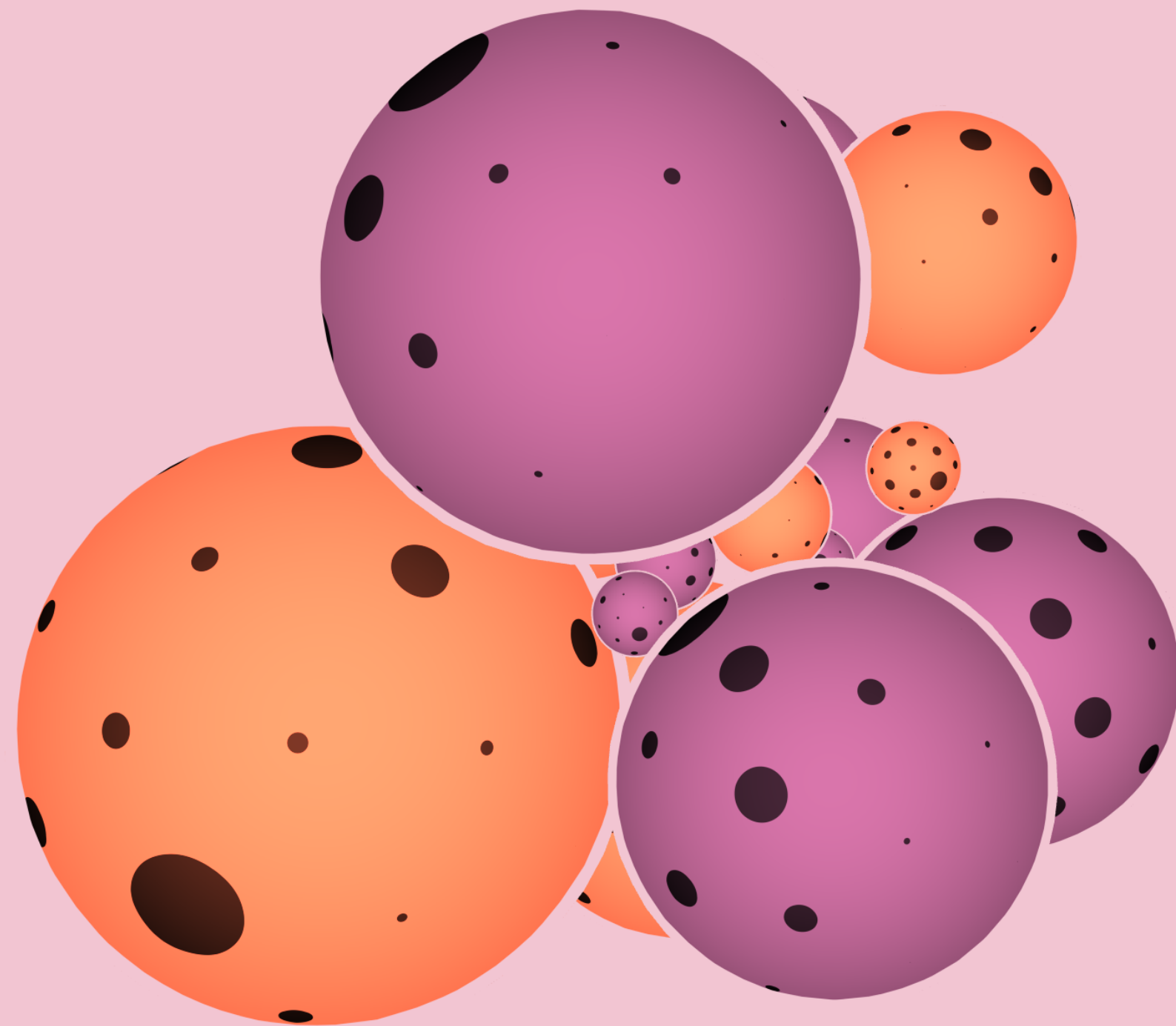








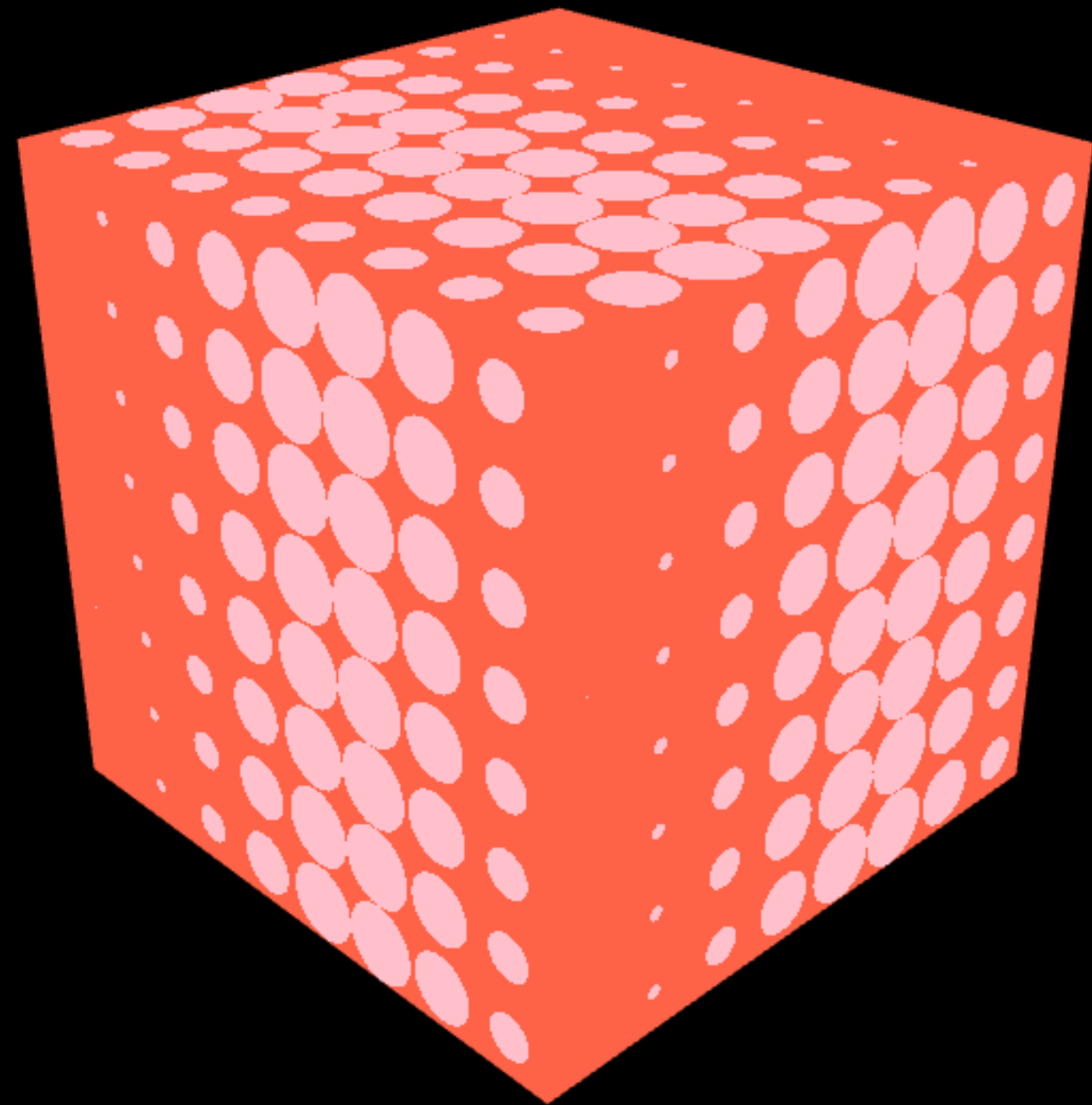






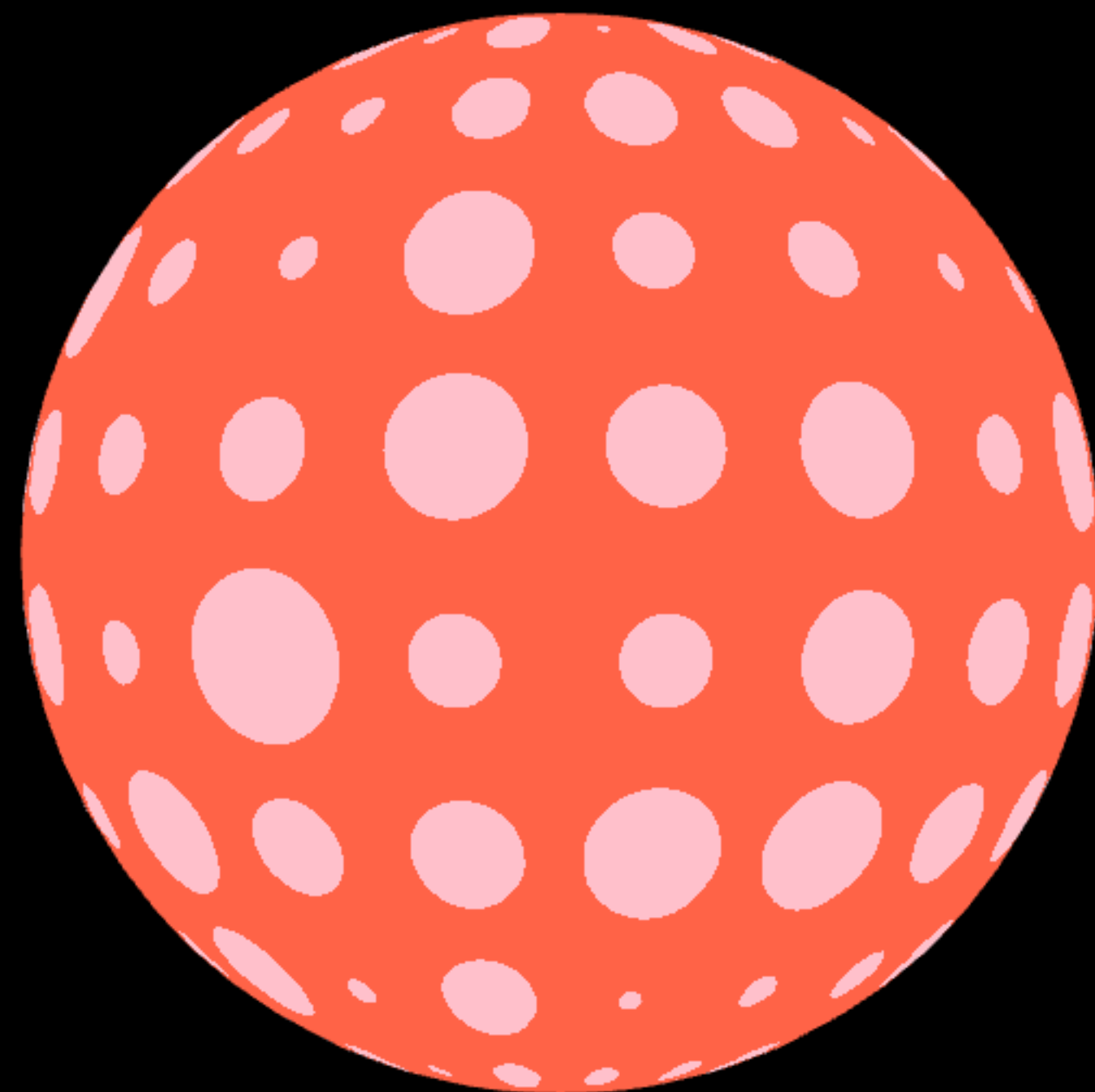
A vector field visualization on a black background. The field is composed of numerous small white line segments representing vectors. These vectors are arranged in a grid-like pattern, with their orientation and length varying across the space. In the upper-left region, the vectors are longer and more widely spaced, while in the lower-right region, they are shorter and more densely packed. The text "START SMALL" is centered in the middle of the image in a white, sans-serif font.

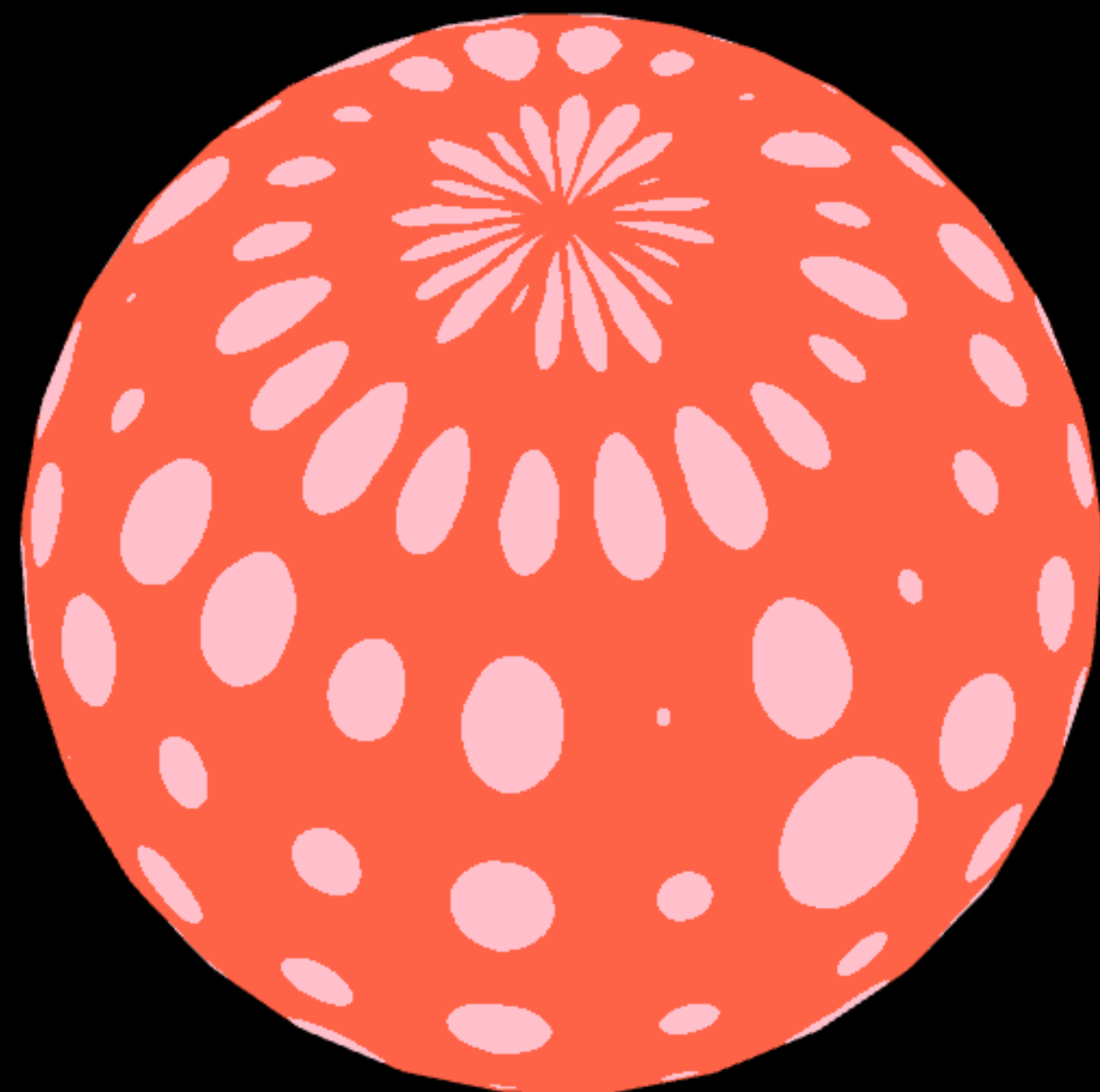
START SMALL

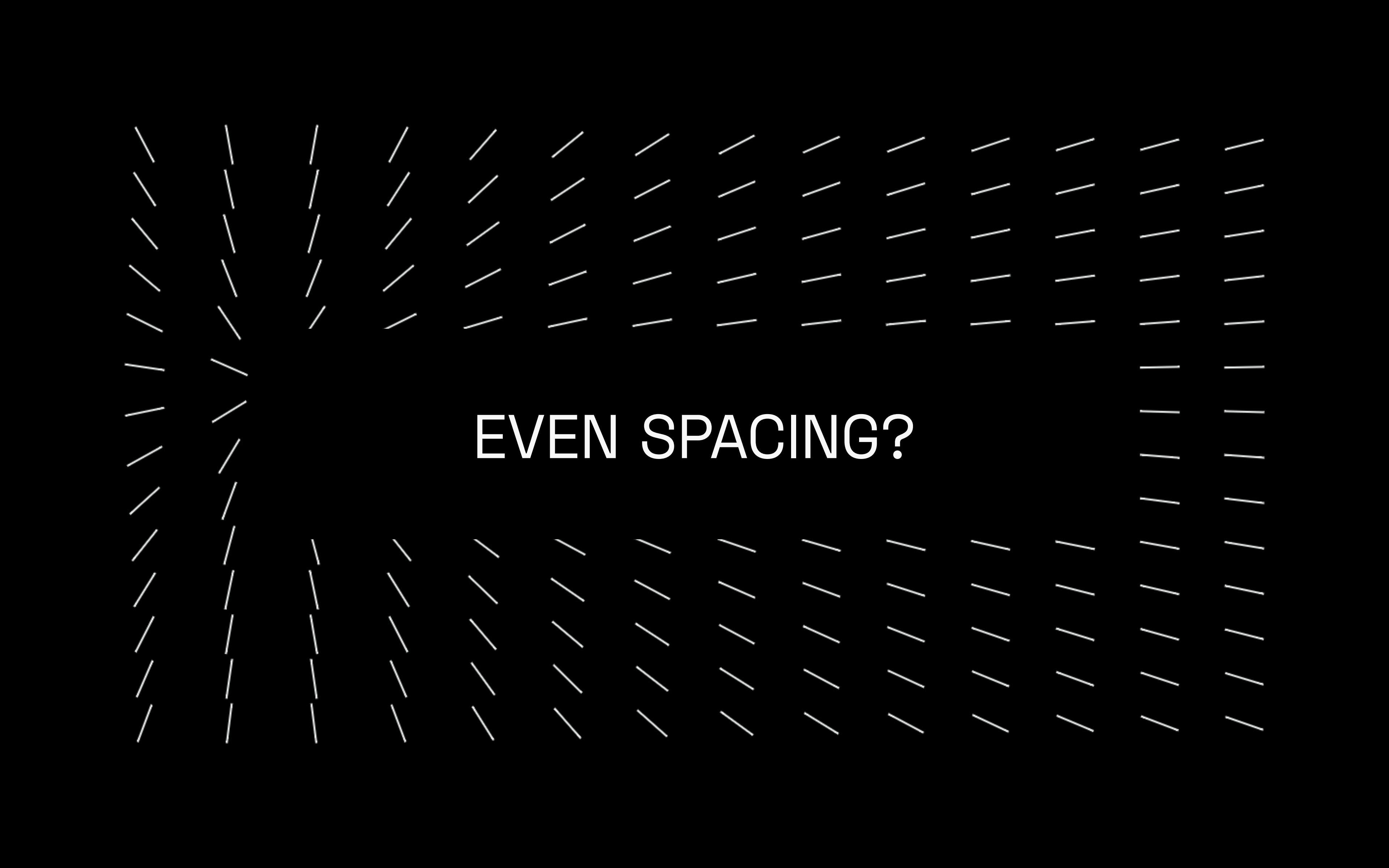




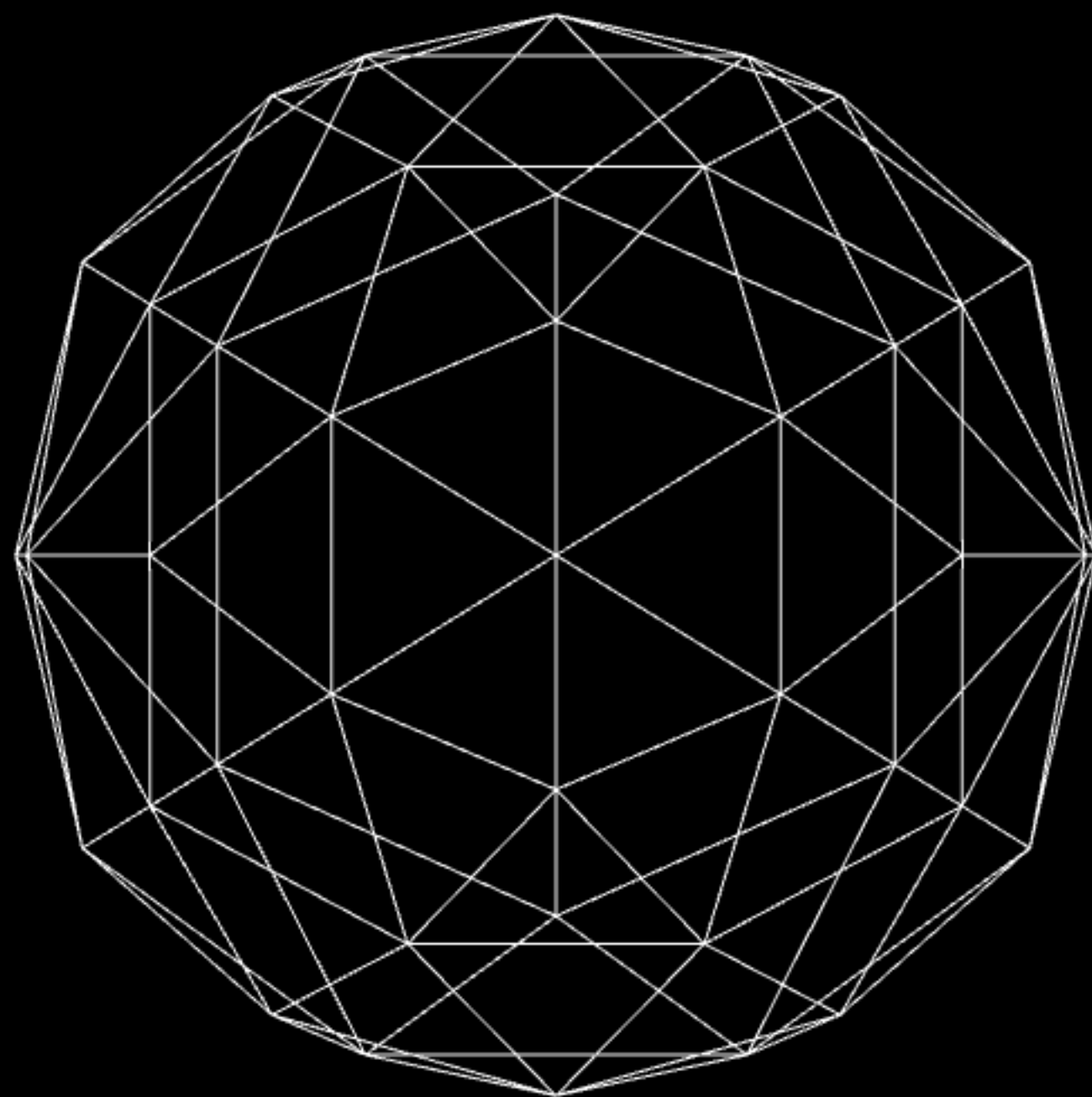
MAKE SOME NOISE



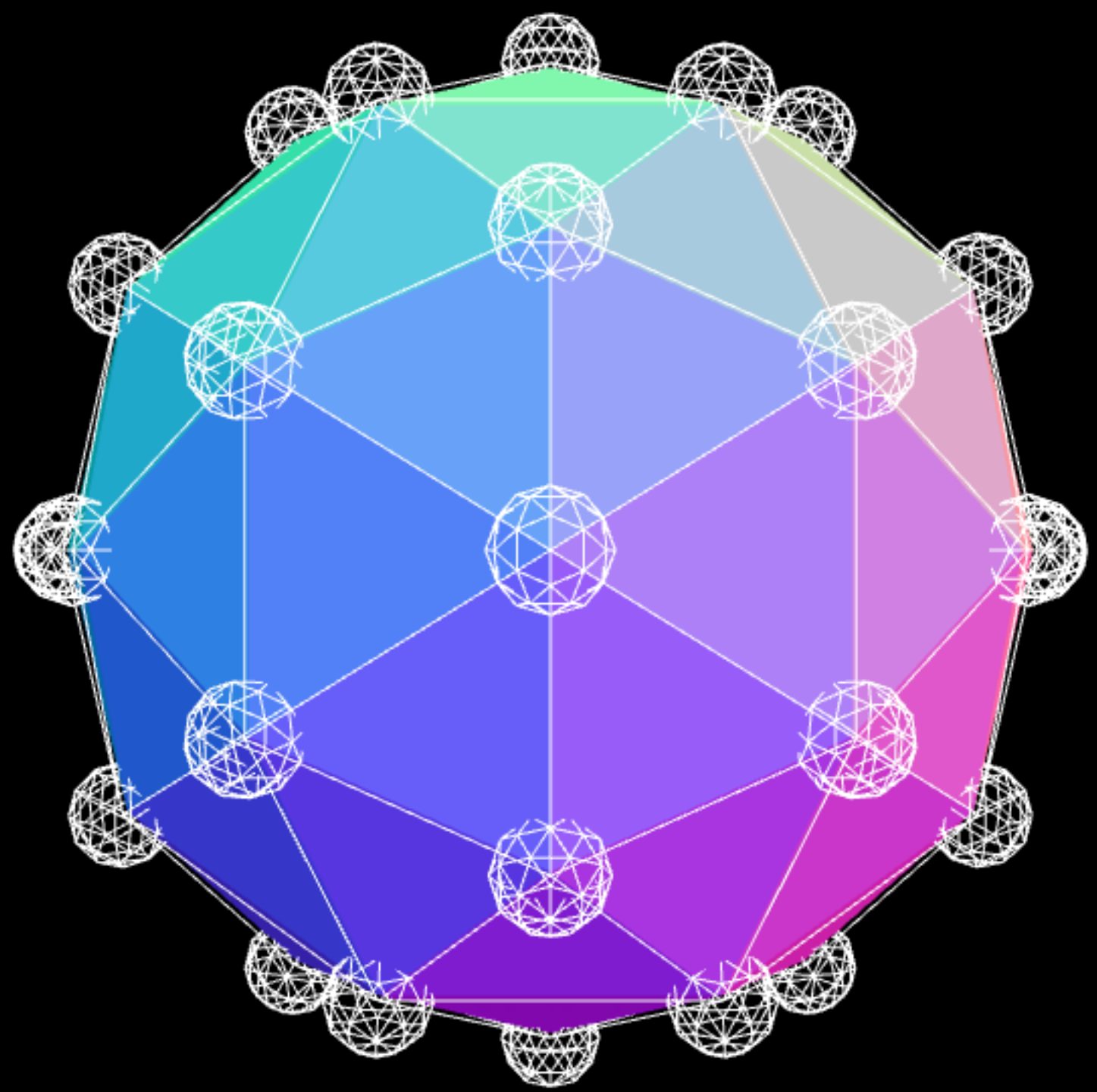


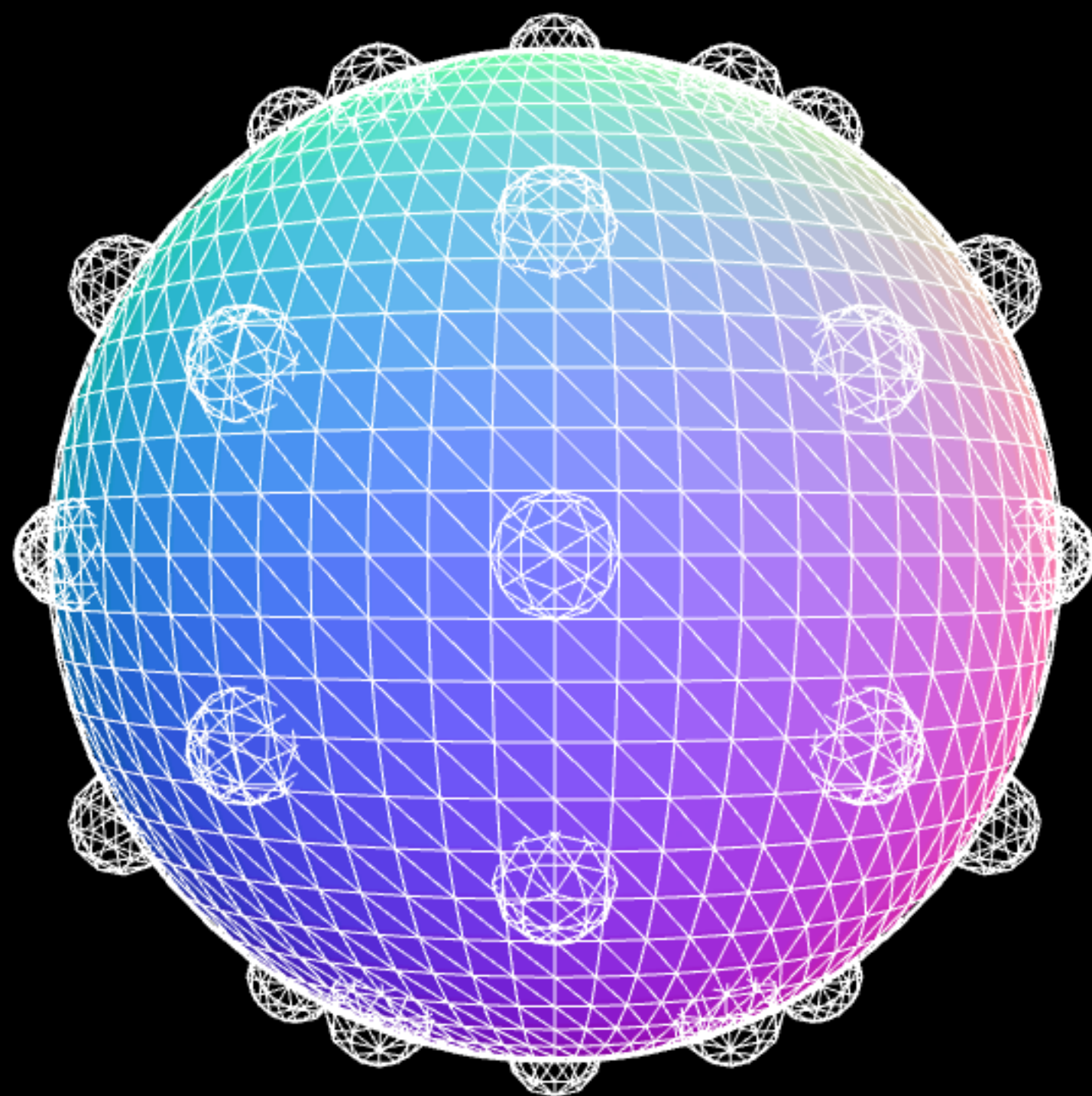
The image features a black background with a grid of white diagonal lines. The lines are arranged in a pattern that is roughly rectangular, with a central area where the lines are absent, revealing the black background. The lines are oriented diagonally, from the top-left to the bottom-right. The spacing between the lines is not uniform, which is the subject of the text in the center.

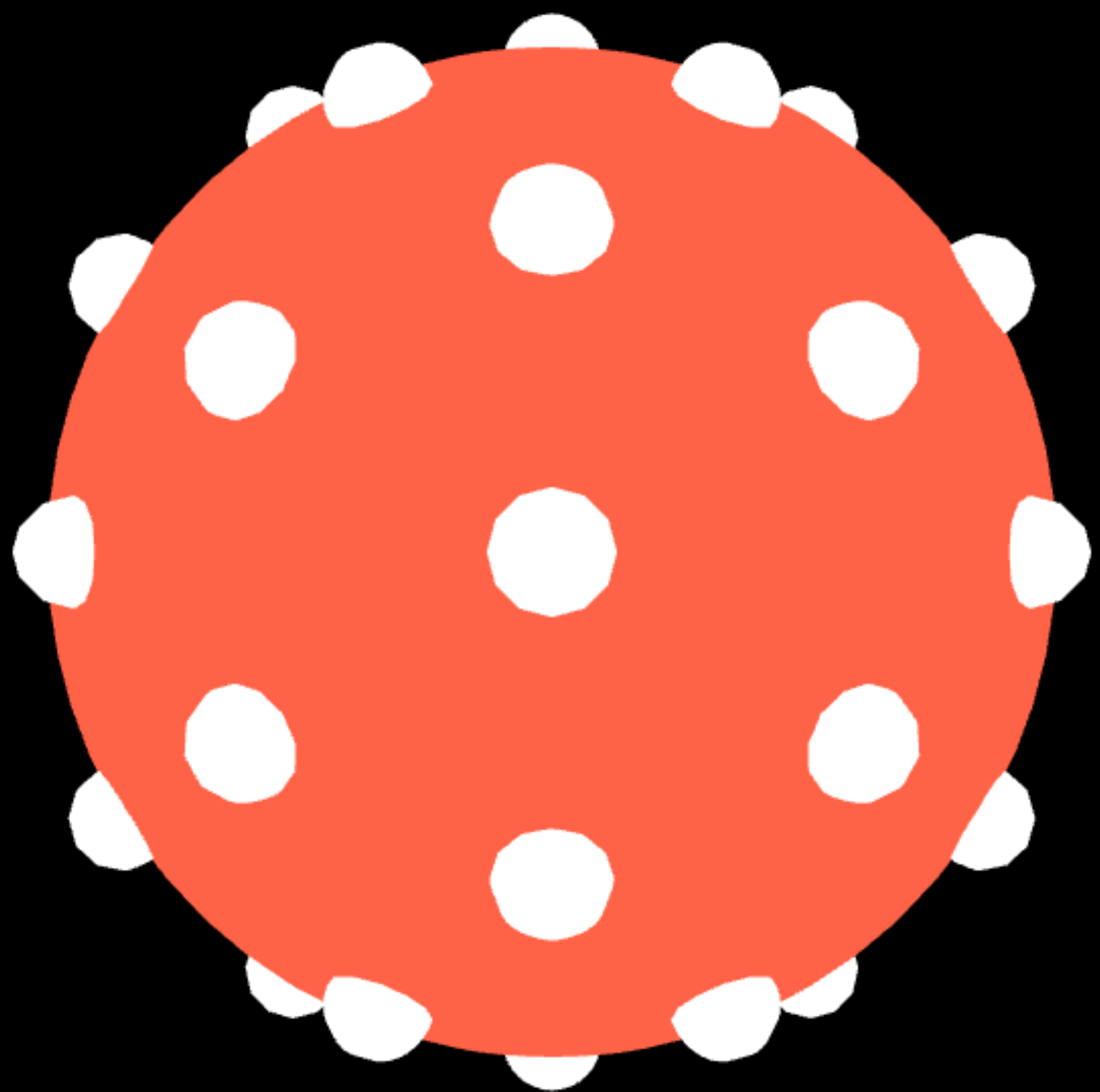
EVEN SPACING?







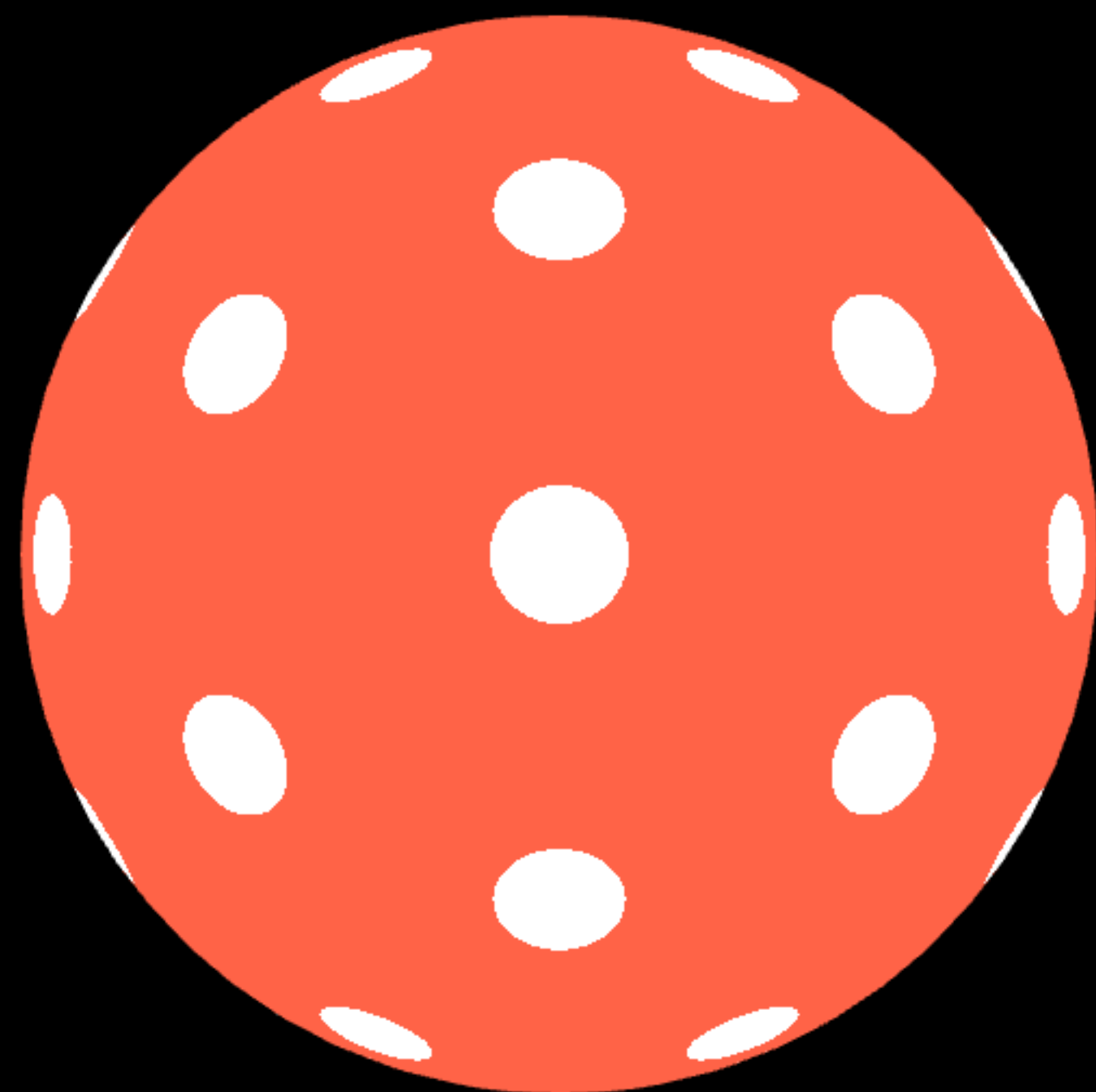


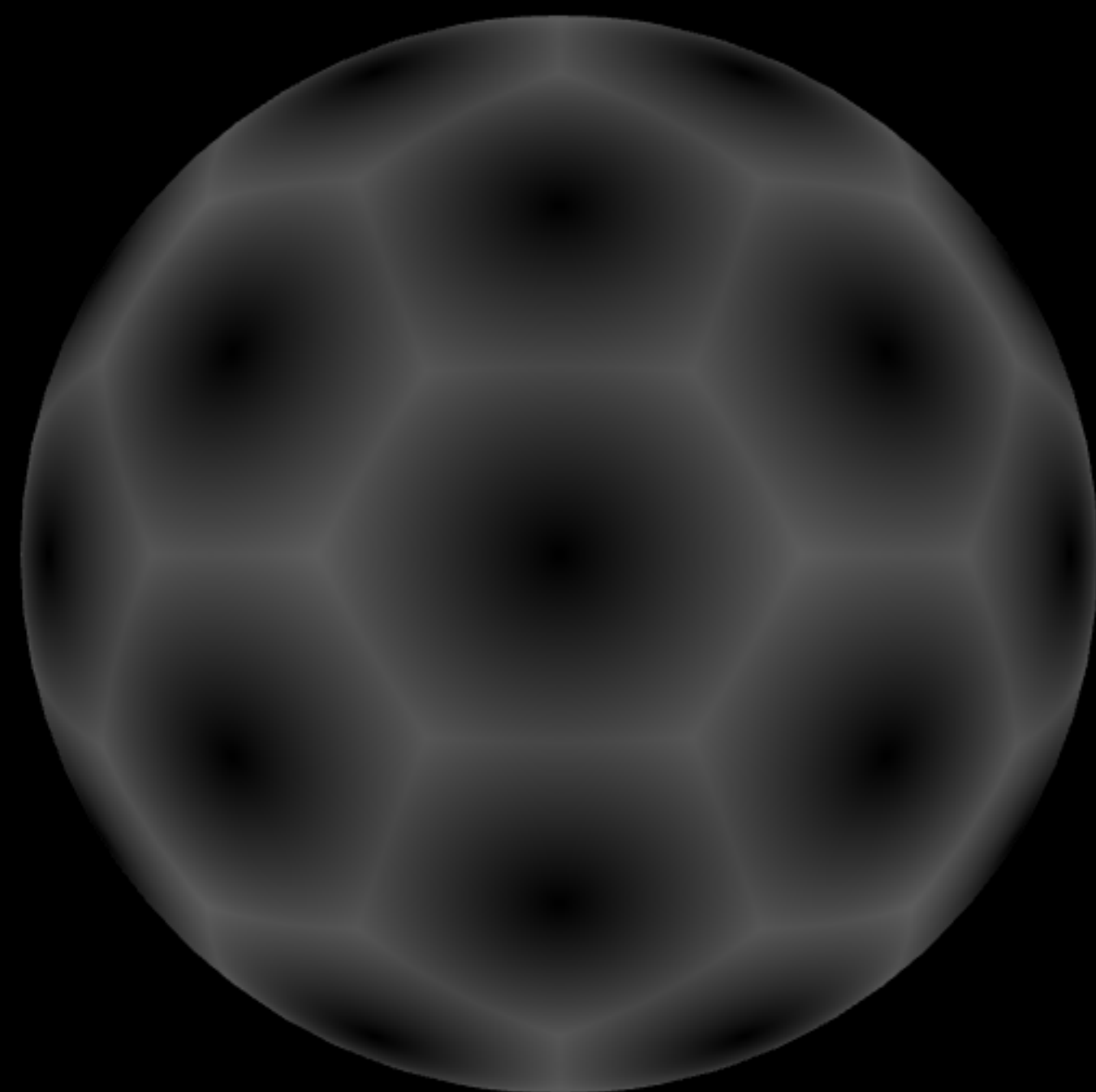






SHADER VERSION

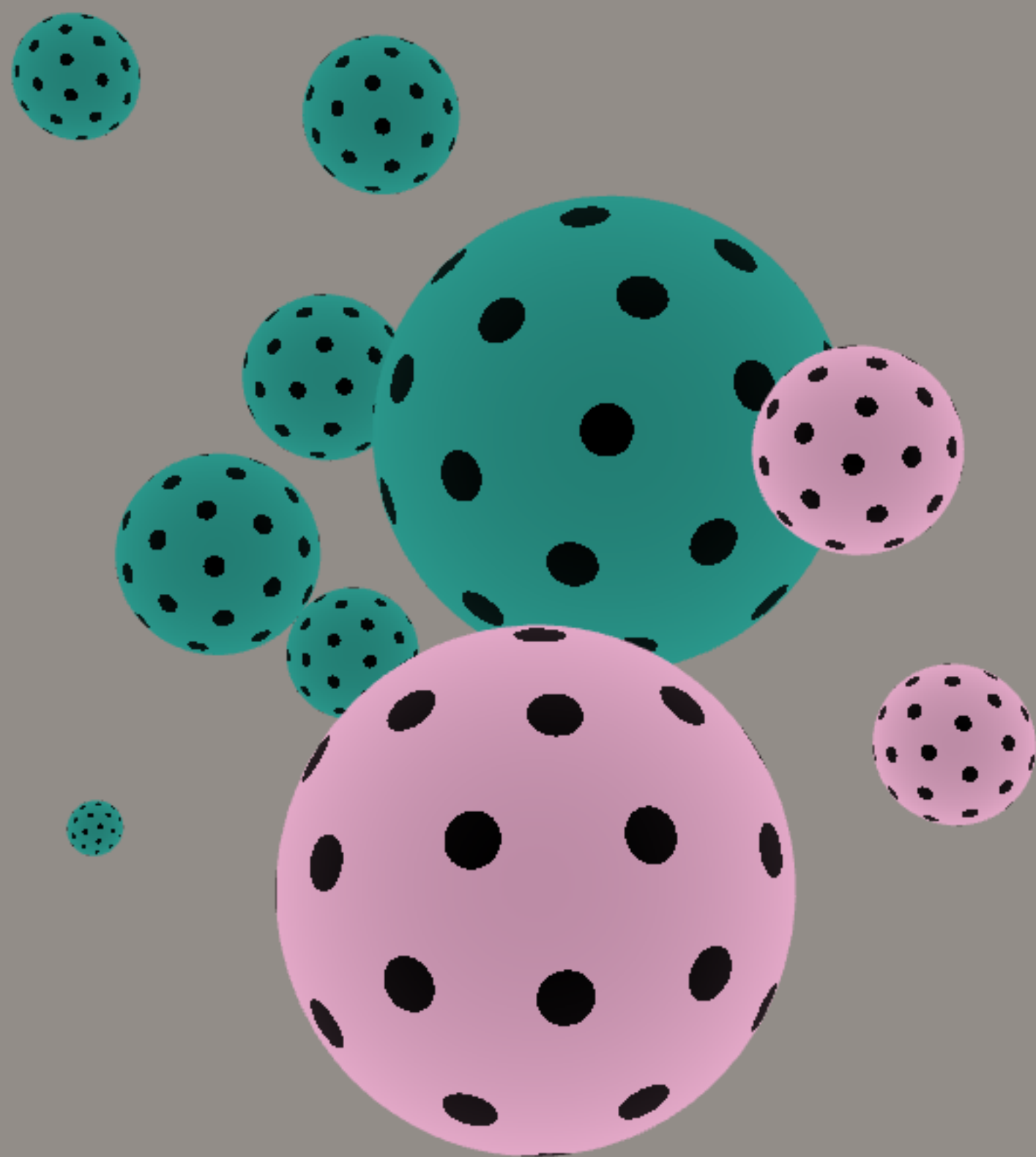






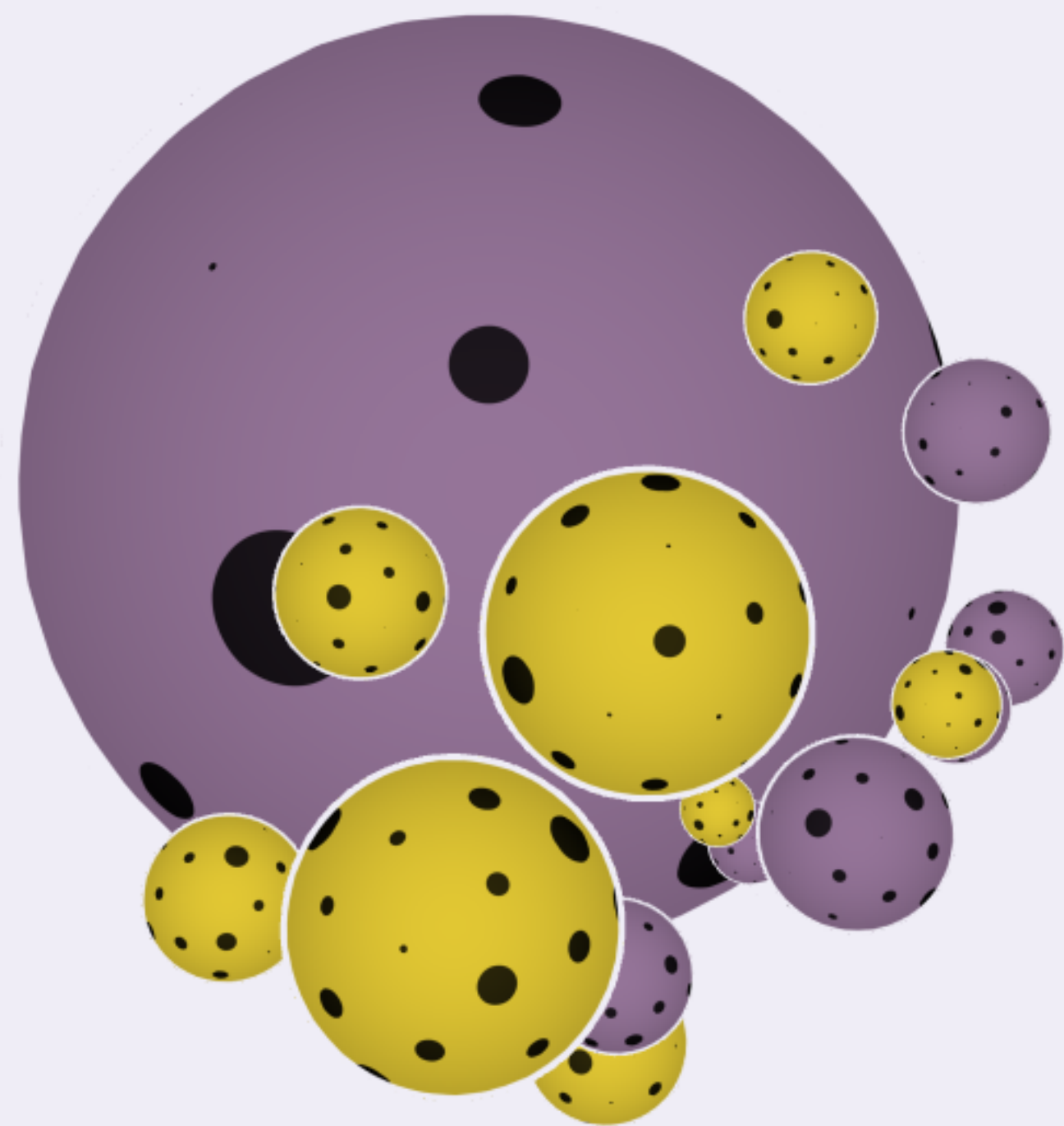


SPRUCE IT UP





OPTIMIZE & POLISH





GOING FURTHER