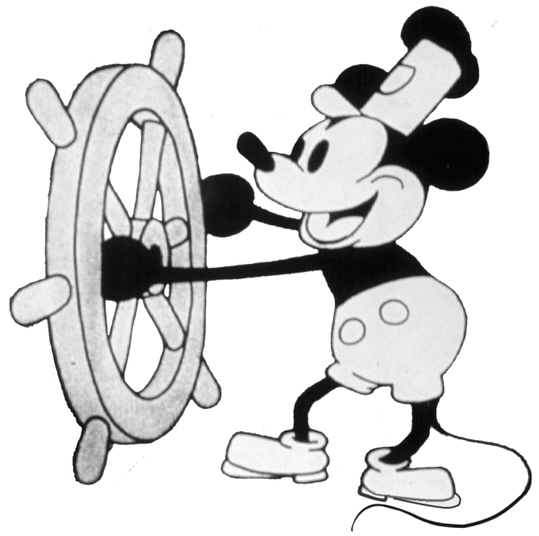


CSCI 461: Computer Graphics

Middlebury College, Fall 2025

Lecture 1: Pixels

Goals for today:

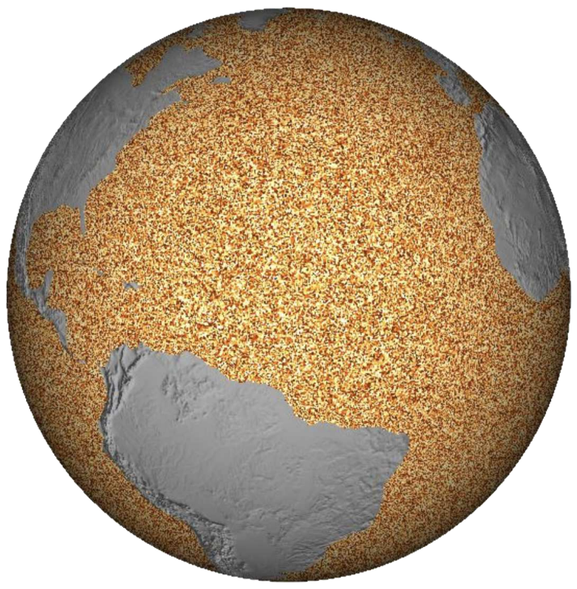
- Introductions!
- The main goal of computer graphics.
- What will we do in this course?

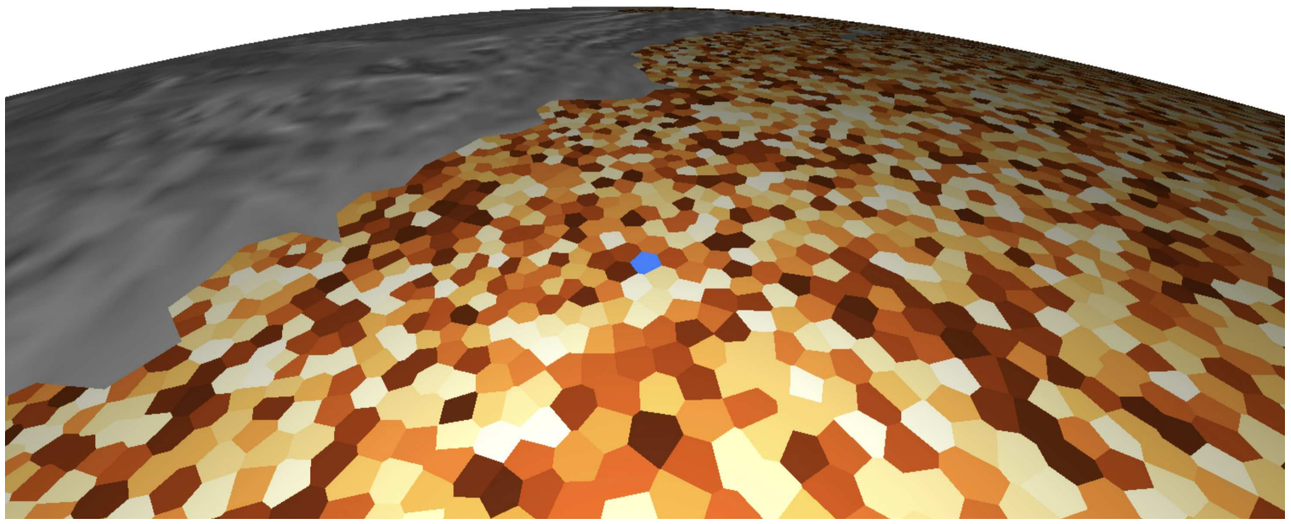
Visit go/cs461, click on **reactions** tab, then click one of the emojis - use this any time! (during class)

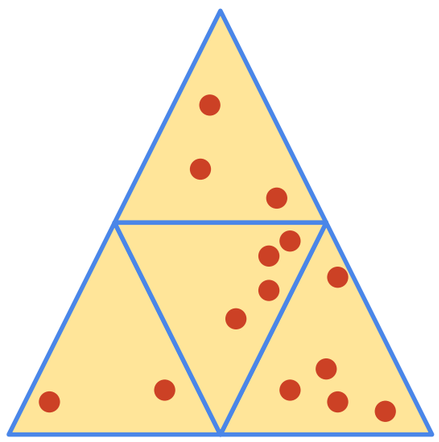
A little about me...

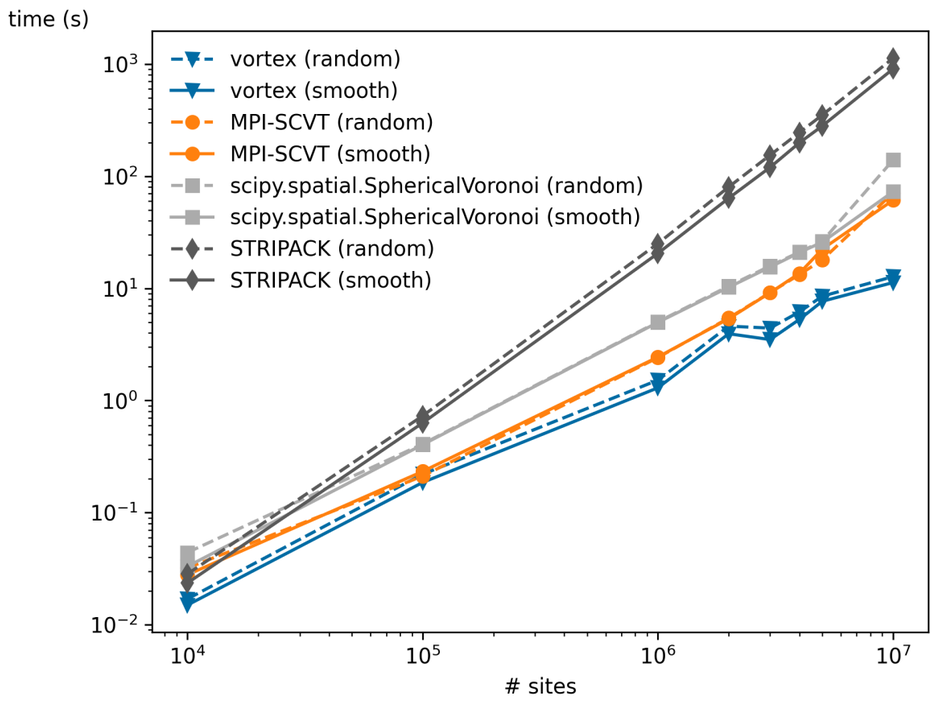
- Please call me Philip.
- I'm going to try to say "zee" for `Z`, but will probably end up saying "zed" a lot.
- My favorite use of computer graphics is in animated movies.
- My research is mostly on geometry and simulations, and computer graphics is a useful tool to visualize results (and debug).

Things I've been working on lately...

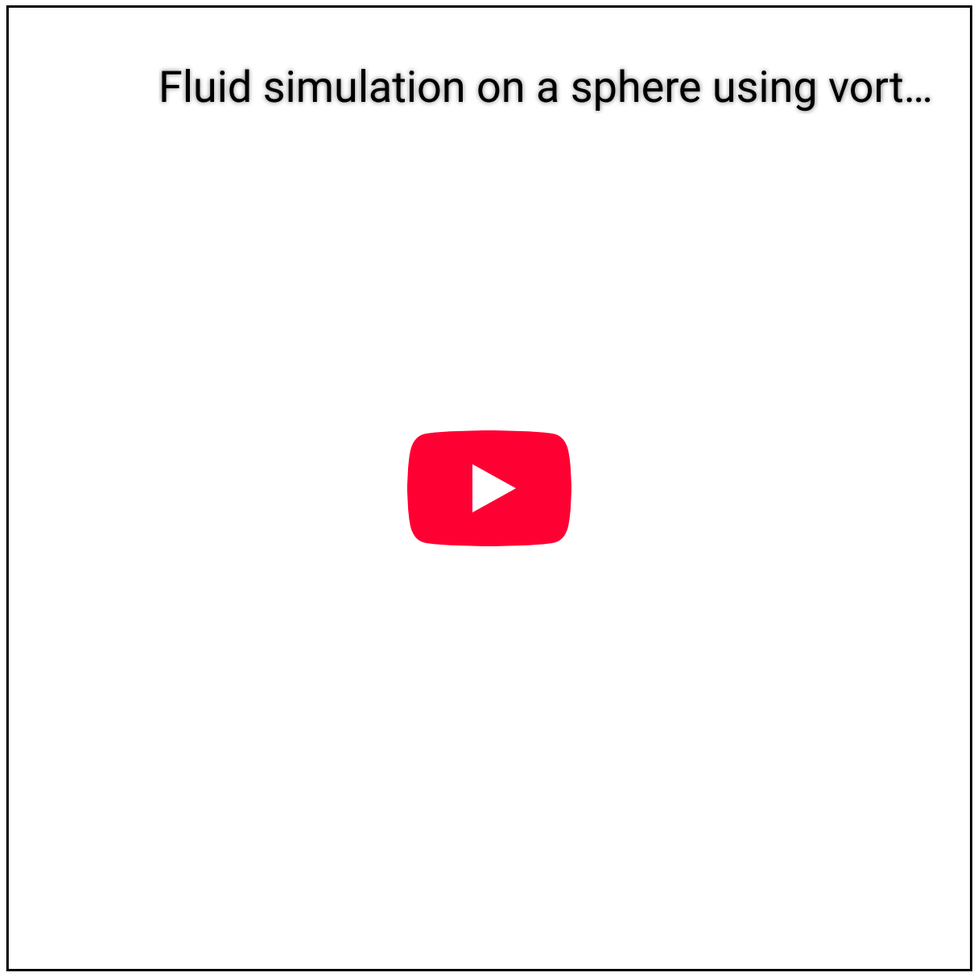








How can we make fluid simulations *faster* (and still accurate)?



(Rightmost video from the `openlb` group)

A little about you!

In groups of 3-4:

- Introduce yourselves!
- What is computer graphics about?
- Where have you seen computer graphics used before?

What is Computer Graphics about?

Computer graphics is about developing computer programs to create visual information.

What this course is NOT.

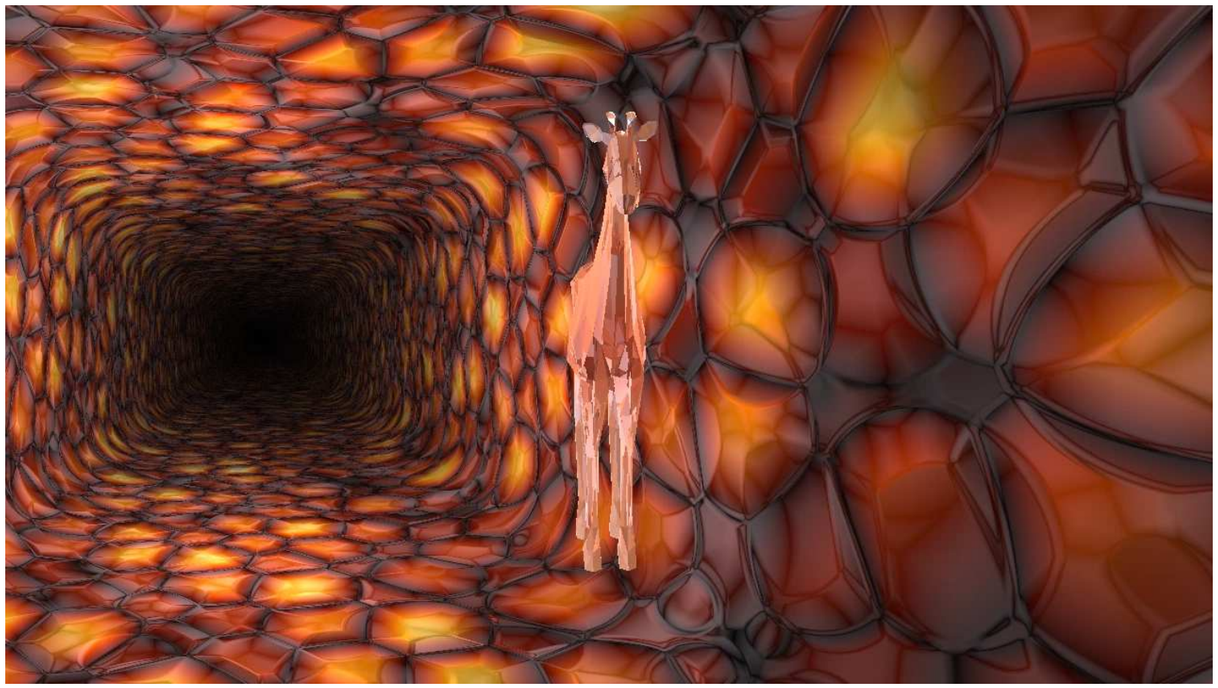
Your job is to develop the graphics technology that artists might need.

By the end of the course you will:

- develop your own ray tracer to render complex scenes and materials,
- display and manipulate three-dimensional models using rasterization techniques (with `WebGL`),
- animate three-dimensional objects and physical systems.

We will use a form of *specification grading*.

- 12 assignments in total: 11 Labs + 1 Project (Shadertoy)
- Labs evaluated using CRN model (similar to EMRN you may have seen before):
  - (N)ot assessable: little to no modification to initial template.
  - (R)evisions required: error or bug.
  - (C)omplete: submission implements all required components.
- Shadertoy Project will build off of an existing Shadertoy.
  - Part 1 (reverse engineering) due in Week 9.
  - Part 2 (extending the Shadertoy) due in Week 12.
  - More details to come.



We will use `GitHub` Classroom to submit assignments.

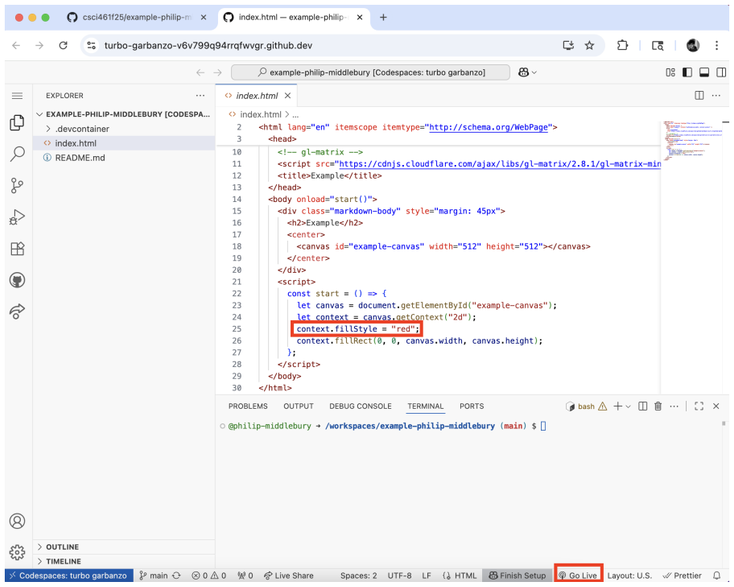
- See setup instructions on course website: https://philipclaude.github.io/csci461f25/setup

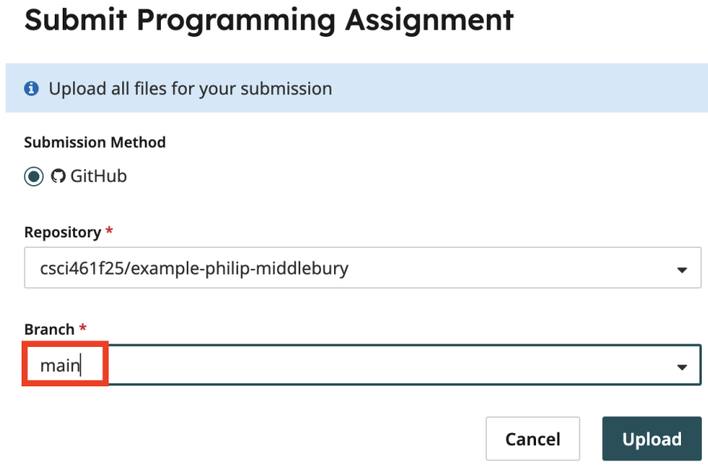
csci461f25-classroom

Accept the assignment — Example

Once you accept this assignment, you will be granted access to the `example-philip-middlebury` repository in the csci461f25 organization on GitHub.

Accept this assignment



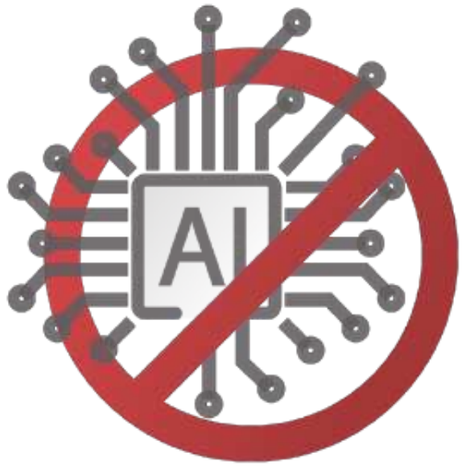


What to expect in this course...

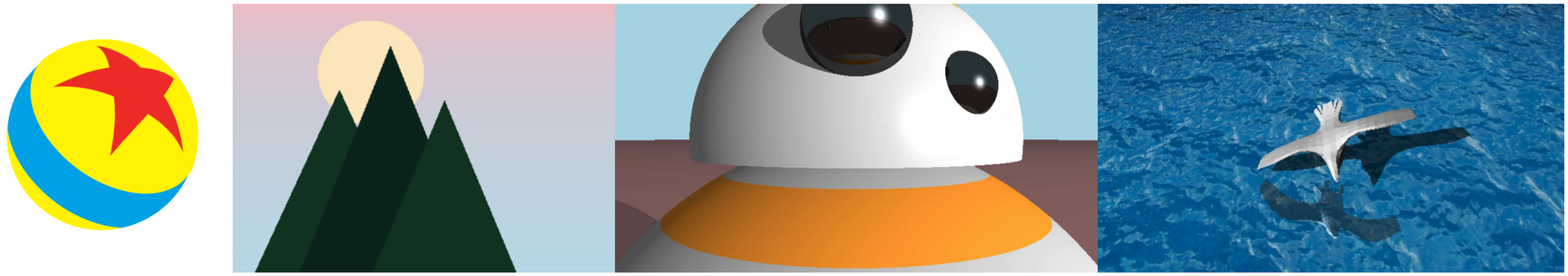
- Lectures and exercises on Tuesdays.
- Please read the notes after class on Tuesday and before the lab on Thursday.
- Pre-lab questions will unlock the lab on Thursday.
- Labs on Thursdays: *Initial* submissions due 1 week after lab period.
  No feedback if submitted after initial due date.
- Feedback on current lab status (CRN), then revise if necessary.
  Final submissions due about 1.5 weeks after initial submission (Tuesdays).
- A LOT of debugging! But also a lot of fun :)

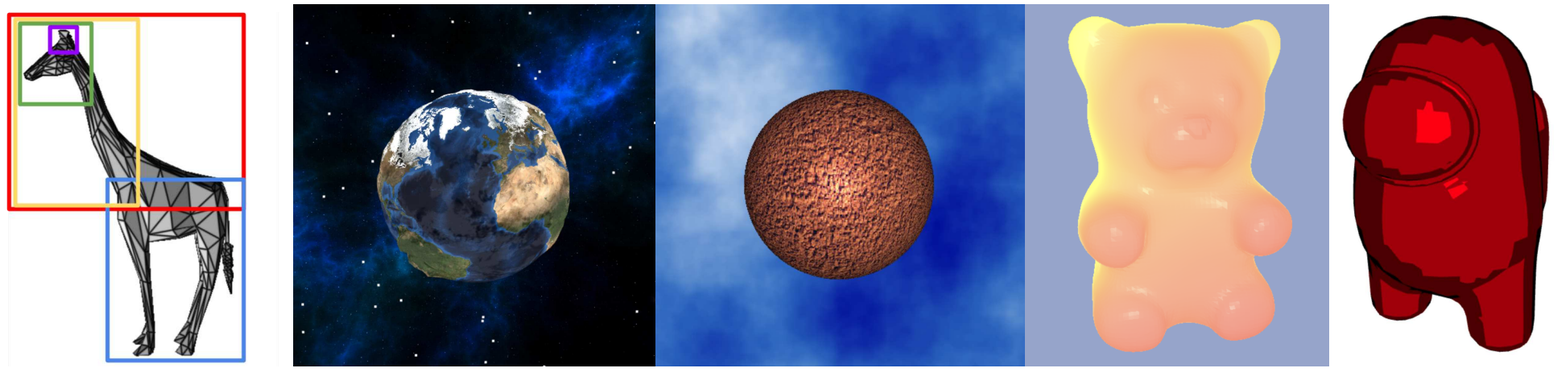
Honor Code and AI policy.

- Labs and Shadertoy Project will be submitted individually, but you are encouraged to work in groups of 2-3 during the lab period on Thursdays.
- You can look at each other's code as you implement the lab and look for bugs.
- You cannot write code for someone else, or let someone else write code for you, or send/receive code in an email or text message.
- You cannot use AI to develop any code for you...except for the Shadertoy project.
- AI is a useful tool to help you understand theoretical concepts but will inhibit your ability to truly understand the material in a way that you can (1) build upon and (2) identify mistakes.
- AI is allowed (and encouraged) for the Shadertoy Project.



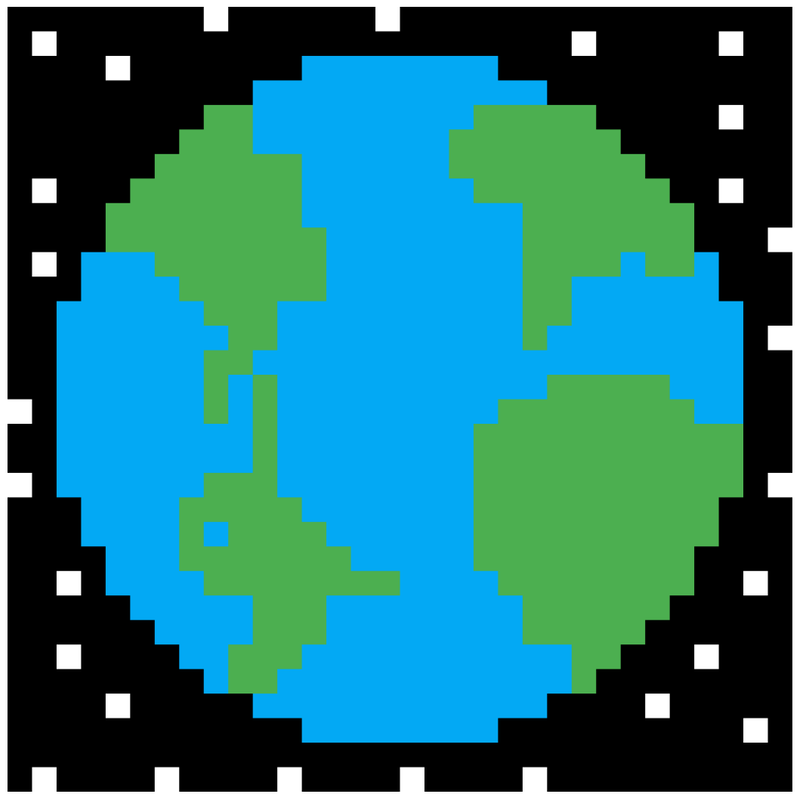
Labs/topics preview.





Let's talk about pixels!

Our goal: assigning pixel colors.



Things to consider:

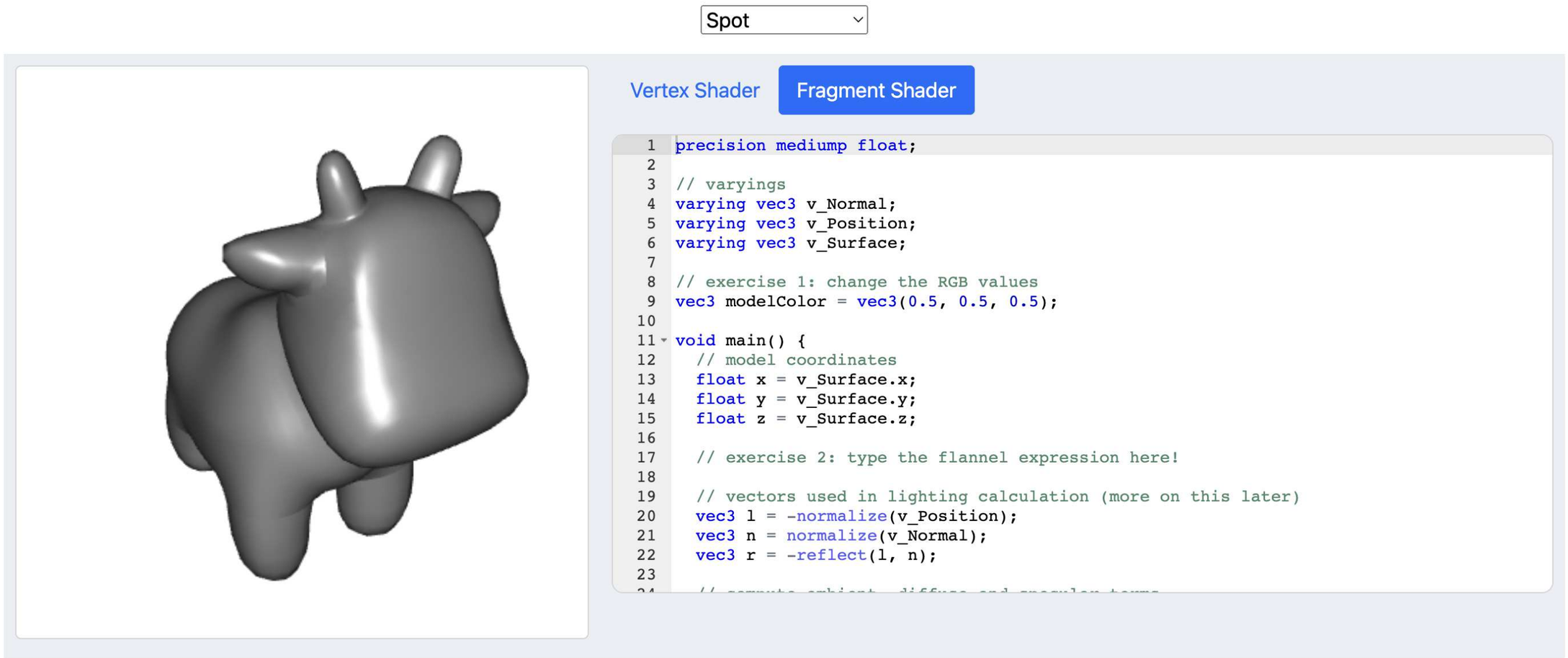
1. What is the **size** of the image?
2. How to represent **color**?
3. What is the **coordinate system** of the image?

We will often represent the color of a pixel using RGB values in between 0 - 1 (sometimes from 0 - 255).

Let's practice with Spot the Cow.

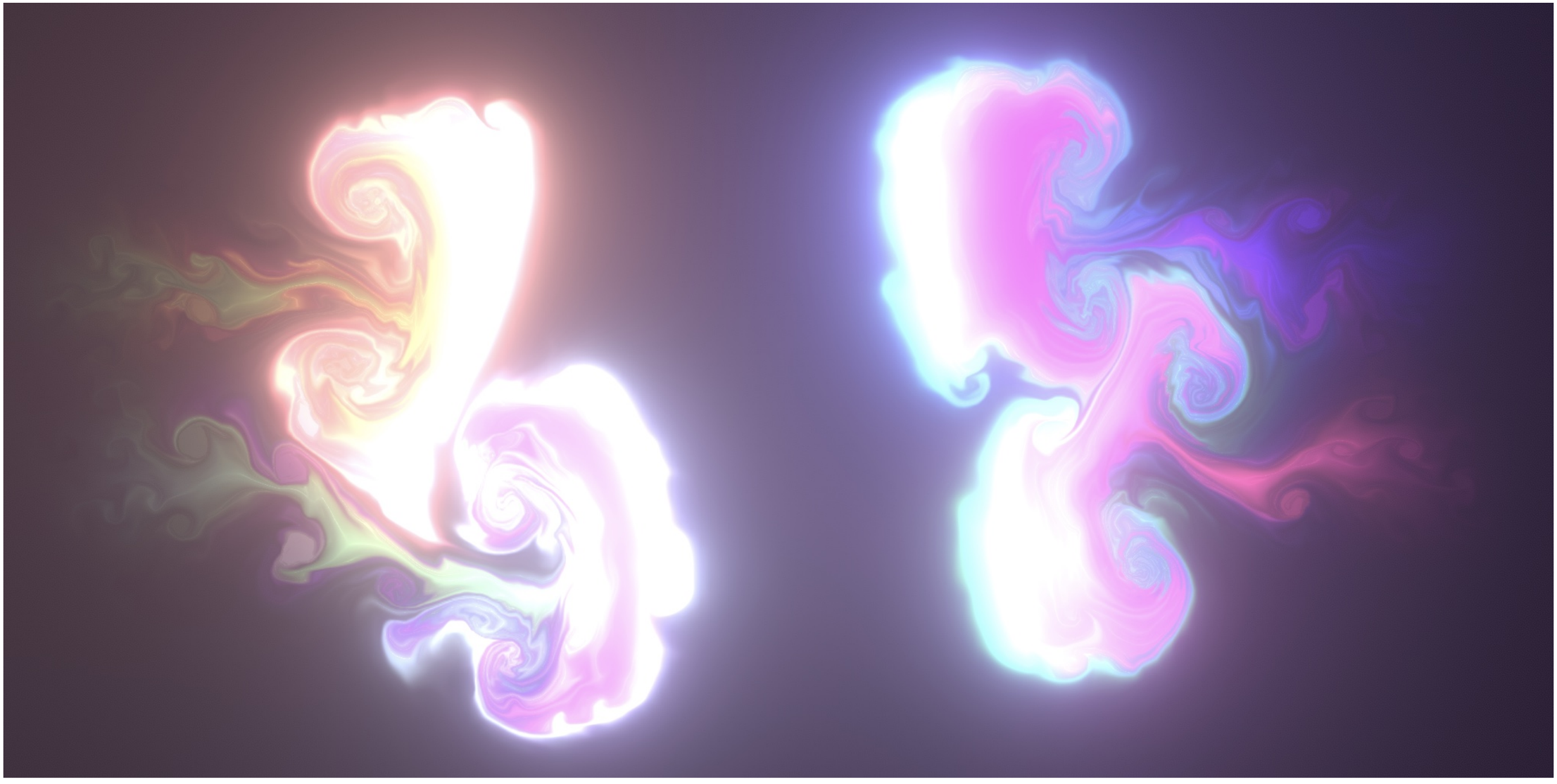
Click to open the shader editor.

(we'll look at **WebGL** and **GLSL** later in the course)



Our goal: assigning pixel colors.

Click to open the WebGL fluids demo.



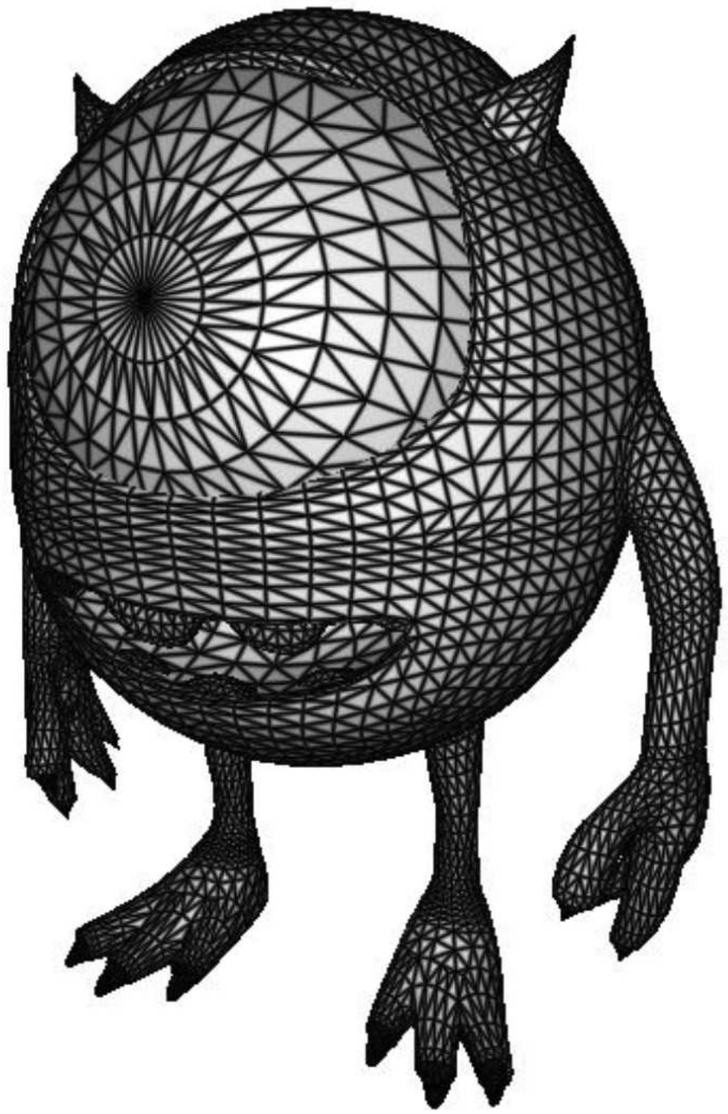
quickly!

JavaScript in one slide.

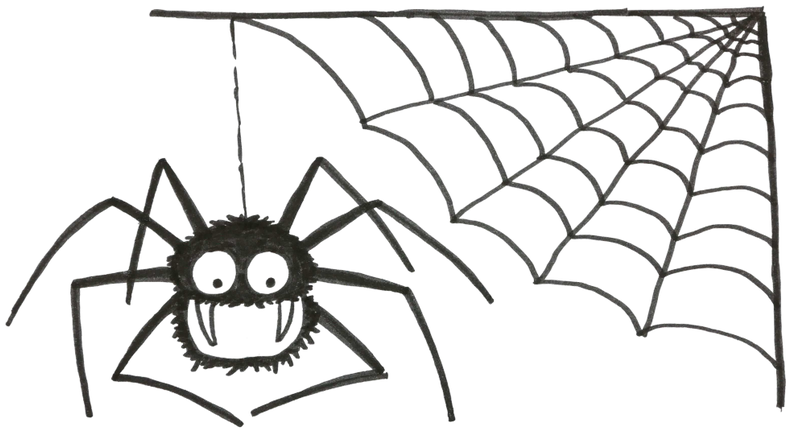
```
class Pixel {
  constructor(r, g, b) {
    this.r = r;
    this.g = g;
    this.b = b;
  }

  scale(a) {
    this.r *= a;
    this.g *= a;
    this.b *= a;
  }
}

Pixel.prototype.set = function(r, g, b) {
  this.r = r;
  this.g = g;
  this.b = b;
}

let p = new Pixel(0.5, 0.5, 0.5); // create a Pixel object
p.scale(255);
p.set(1, 0, 0); // set to red

let image = []; // create a 200 x 100 image
for (let i = 0; i < 200; i++) {
  for (let j = 0; j < 100; j++) {
    const r = Math.random();
    const g = Math.random();
    const b = Math.random();
    let p_ij = new Pixel(r, g, b);
    if (i * i + j * j < 50 * 50) {
      p_ij.set(1, 0, 0);
    }
    image.push(p_ij);
  }
}
```

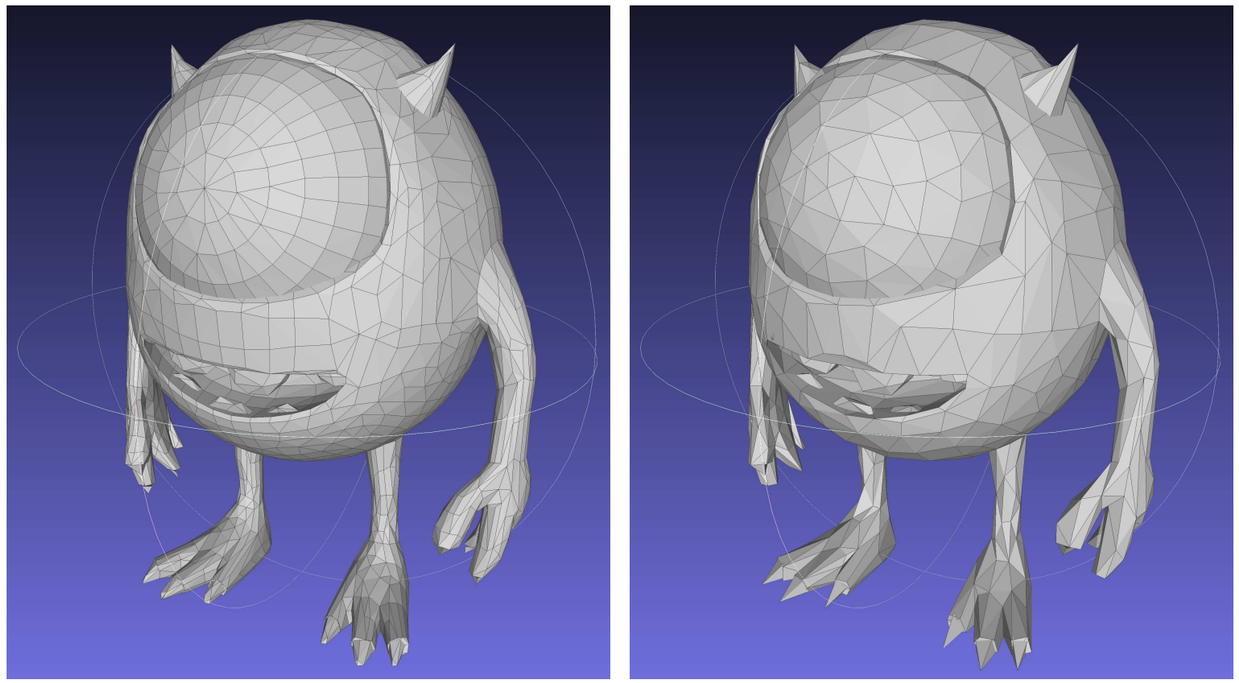
How would you represent the geometries in the previous examples?

We can use a "soup" of triangles in which we store three 3d points per triangle.



What if we want to modify our models?





We need a concept of "connectedness" (more on this later in the semester, some details in the Chapter 1 notes).

One final note: please come to class.

The word COMPUTER means...

Final grade modifiers (+/-) will be used for attendance. But if you need to miss class, please submit this form to let me know.

See you on Thursday!

- Please complete the Introduction Survey.
- Familiarize yourself with syllabus, calendar, notes from today:

  go/cs461
- Go through steps on Setup page on course website.
- Review `JavaScript` (see links in Chapter 1 notes).
- Office hours: Mondays 10am - 11am (my office, Room 219) and Thursdays 2:30pm - 4:30 (Room 224).