

计算机图形学 Project1 说明文档

20302010008 林奕铨

项目目录及文件说明

<code>readme.pdf</code>	说明文档
<code>config.js</code>	配置文件 (与要求相同, 未做改动)
<code>scanConversion.html</code>	项目 HTML 文件 (仅包含网页框架)
<code>scanConversion.js</code>	项目 JavaScript 文件 包括: 类、方法等全部扫描转化逻辑

开发及运行环境

开发环境: macOS Ventura, Visual Studio Code, Safari 16.3

运行环境: Safari 16.3

运行及使用方法

运行方法: 使用支持 `HTML5` 的浏览器打开 `scanConversion.html` 即可。

使用方法: 使用鼠标拖动多边形的顶点, 查看多边形网格的动态绘制情况。

项目亮点

▼ 架构清晰、注释完整

本项目使用了 `class` 关键字语法糖, 以提供更直观的项目架构。

项目共有三个类, 由小及大分别为 `Point`, `Polygon`, `Canvas`

▼ Class Point

- ▶ 成员: `x, y, z` - 三维坐标分量 (本项目中 $z = 0$)
- ▶ 方法: `draw(ctx, r)` - 在给定上下文中以给定半径作点

▼ Class Polygon

- ▶ 成员: `points` - `Point` 对象构成的数组, 按序存储多边形的所有顶点
- ▶ 成员: `color` - `[r, g, b, a]` 构成的数组
- ▶ 方法: `fill(ctx)` - 使用扫描线方法在给定的上下文中作多边形

▼ Class Canvas

- ▶ 成员: `points` - `Point` 对象构成的数组, 存储画布上所有可拖动点
- ▶ 成员: `polygons` - `Polygon` 对象构成的数组, 存储画布上所有多

- ▶ 方法: `clear_and_draw` - 清空画布, 调用 `points` 中所有对象的 `draw` 方法和 `polygons` 中所有对象的 `fill` 方法。
- ▶ ... 其余辅助成员和方法 ...

除此之外, 项目中还有两个函数。

- ▶ `initCanvas()` - 使用 `config.js` 初始化画布 (包括添加点、添加多边形、绘制画布)
- ▶ `addDragHandler()` - 向画布添加拖动事件处理程序, 当一个点被拖动时, 画布将被重绘。

为了使源代码文件更清晰, 项目使用了 `JSDoc` 为类和函数添加了注释, 使阅读更为容易。

▼ 拖动操作的监听与处理

DOM中没有提供“拖动操作监听器”, 因此需要通过 `mousedown`, `mousemove` 和 `mouseup` 实现对于拖动的处理。

▼ `mousedown`

当鼠标按下时, 会遍历 `canvas` 的 `points`, 查看是否有一个 `point` 对象在鼠标点击位置的范围内, 如果有, 则将 `drag_flag` 置为 `true`, `drag_point` 设置为该点, `drag_offset` 设置为鼠标点击坐标到 `drag_point` 坐标的偏移量。

▼ `mousemove`

当鼠标移动时, 如果 `drag_flag` 为 `true`, 则表明正在拖动。通过计算偏移量来更新 `drag_point` 的位置, 并重绘 `canvas`。

▼ `mouseup`

当鼠标释放时, 将 `drag_flag` 置为 `false`, `drag_point` 清空。

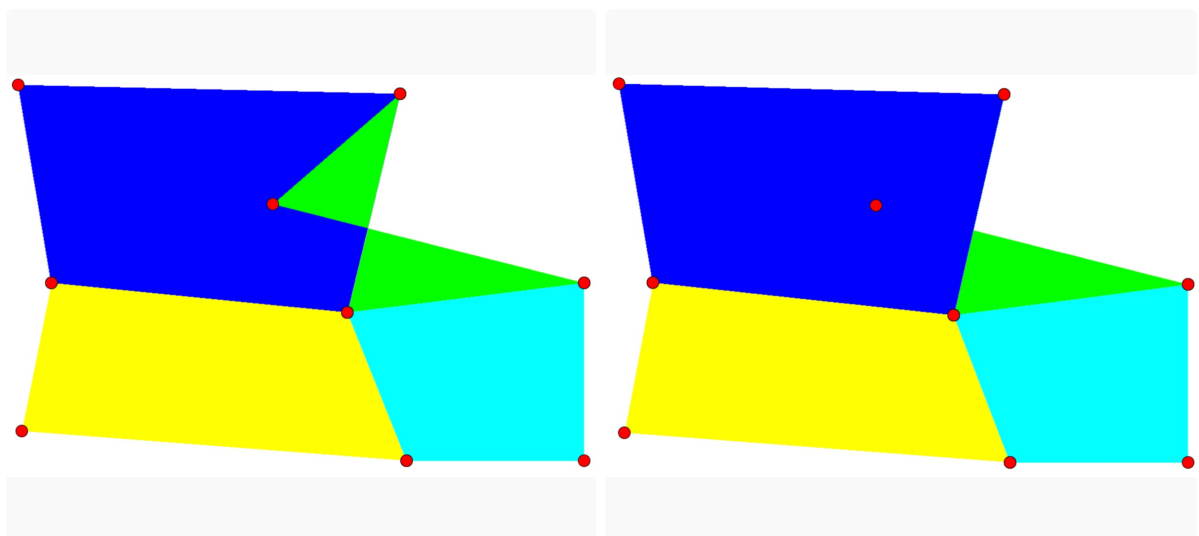
| 开发中遇到的问题及解决方法

▼ 多边形渲染时的覆盖问题

▼ 问题描述

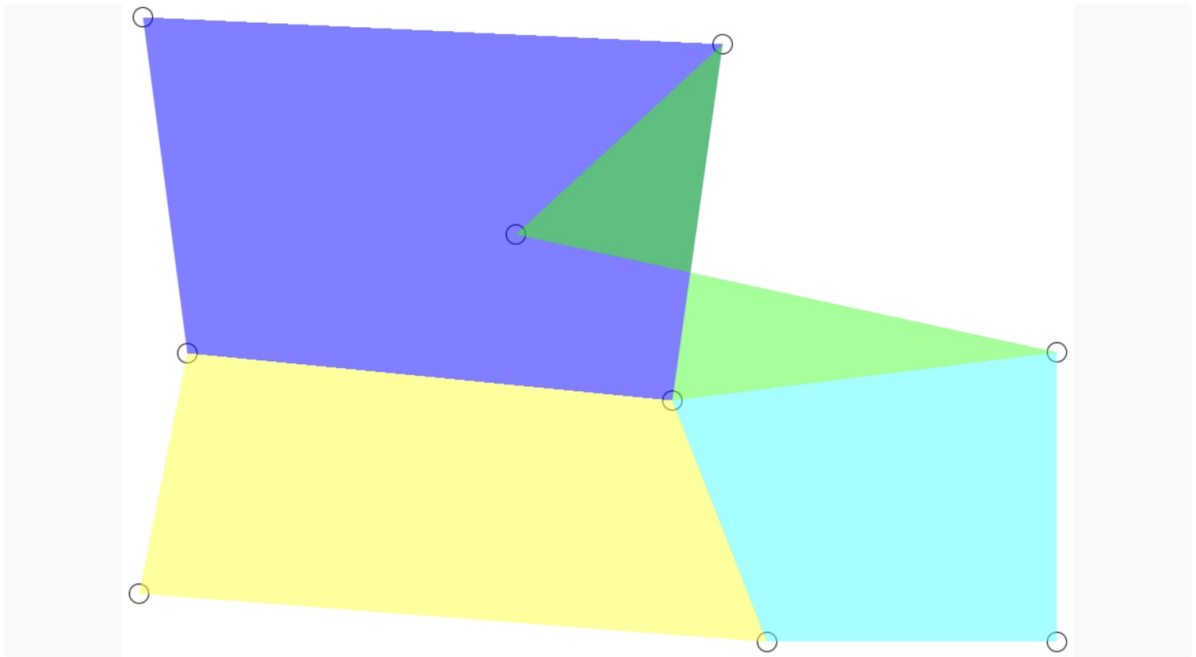
在项目给的Demo中, 多边形颜色的 `alpha` 分量被设为 `a=1` (即完全不透明)。这导致在多边形渲染中可能会造成遮蔽, 又因为渲染先后顺序的不确定, 将导致不同的遮蔽方式。

如下图所示, 左右两图的顶点坐标相同, 但渲染结果不同。



▼ 问题解决

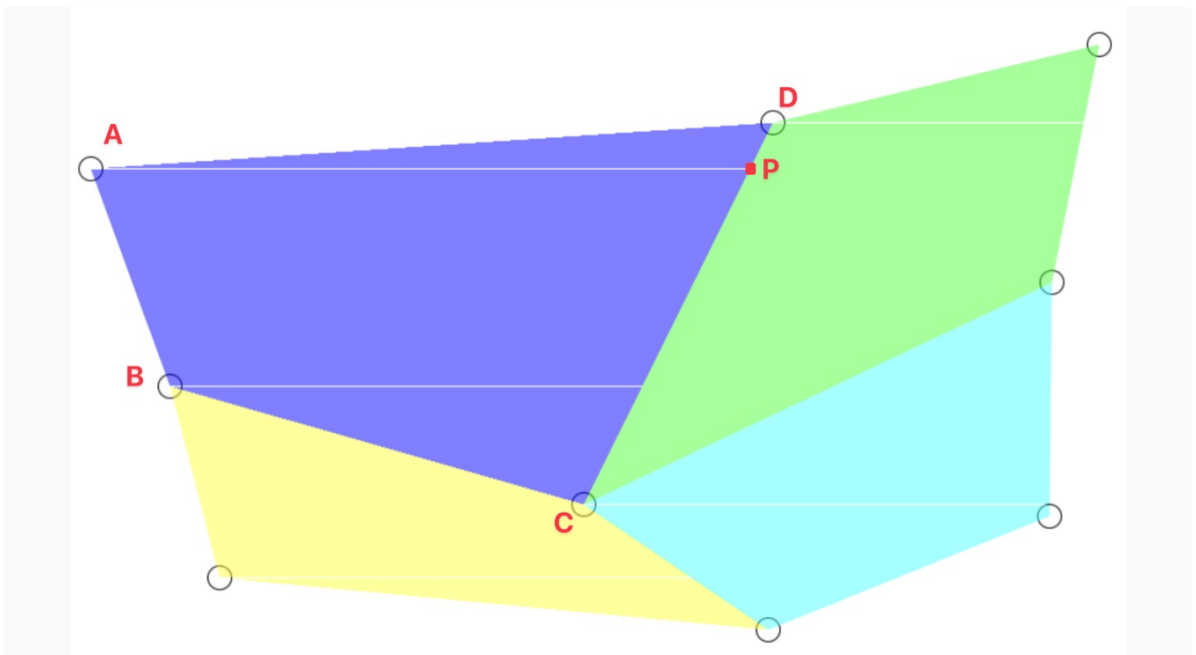
将多边形颜色的 `alpha` 分量调整为常量 `ALPHA` (默认为 0.5), 使得多边形重叠部分的颜色进行叠加 (而非遮蔽)。



▼ 扫描线算法的边界处理问题

▼ 问题描述

当扫描线的纵坐标与多边形顶点的纵坐标相同时, 可能会发生填色异常问题。如下图中的白线所示, 应填色的部分没有被上色。



▼ 问题解决

出错原因在于对于边界情况的处理。

以图中的四边形 ABCD 为例，过点 A 的扫描线与边 AB、边 AD、边 CD 均有交点 (分别为点 A、点 A、点 P)。此时扫描线算法在点 A 与点 A 之间填充了颜色、点 P 没有与其他点配对，因此 AP 之间没有填充颜色。

将边的定义调整为 [Point1, Point2) ($\text{Point1.y} < \text{Point2.y}$) 即可解决上述问题。

| 项目中可能存在的缺陷

在本项目中，若拖动一个顶点，则会引发整张画布的重绘。在仅有 9 个顶点、4 个多边形时，渲染性能尚未出现明显问题。然而，随着数据的复杂度增加，可能会存在性能问题，其中一种解决方式是：在拖动某个顶点时，仅重绘与之相关的多边形。