郭隆邦 2023-03-03

7. 抗锯齿后处理

three.js提供了多种抗锯齿的后处理，下面给大家演示下`FXAAShader`和`SMAAPass`两种抗锯齿的后处理。

gltf模型使用后处理带来的锯齿

你可以对比下面两种情况的渲染效果。(使用后处理之后带来了锯齿)

- gltf工厂模型，无后处理（课件文件：“无后处理的gltf工厂”）
- gltf工厂模型，有后处理`EffectComposer`(课件文件：“演示”)

调整相机参数，方便观察工厂设备B的渲染细节

```js
// camera.position.set(202, 123, 125);
camera.position.set(1.1, 11.8, 62.4);
// 可视化选择相机位置
controls.addEventListener('change',function(){
    console.log('camera.position',camera.position);
})
```

FXAA抗锯齿通道

FXAA减弱了锯齿，但是并不完美。

```js
// ShaderPass功能：使用后处理Shader创建后处理通道
import {ShaderPass} from 'three/addons/postprocessing/ShaderPass.js';
// FXAA抗锯齿Shader
import { FXAAShader } from 'three/addons/shaders/FXAAShader.js';
```

```js
// 设置设备像素比，避免canvas画布输出模糊
renderer.setPixelRatio(window.devicePixelRatio);
```

`.getPixelRatio()` 获取 `renderer.setPixelRatio()` 设置的值

```js
// .getPixelRatio()获取设备像素比
const pixelRatio = renderer.getPixelRatio();
```

设置 `FAXX` 抗锯齿通道

```js
const FXAAPass = new ShaderPass( FXAAShader );
// `.getPixelRatio()`获取`renderer.setPixelRatio()`设置的值
const pixelRatio = renderer.getPixelRatio();//获取设备像素比
// width、height是canva画布的宽高度
FXAAPass.uniforms.resolution.value.x = 1 /(width*pixelRatio);
FXAAPass.uniforms.resolution.value.y = 1 /(height*pixelRatio);
composer.addPass( FXAAPass );
```

SMAA抗锯齿通道

SMAA相比较FXAA抗锯齿效果更好一些。

```js
// SMAA抗锯齿通道
import {SMAAPass} from 'three/addons/postprocessing/SMAAPass.js';
```

创建 `SMAAPass` 抗锯齿通道

```js
//获取.setPixelRatio()设置的设备像素比
const pixelRatio = renderer.getPixelRatio();
// width、height是canva画布的宽高度
const smaaPass = new SMAAPass(width * pixelRatio, height * pixelRatio);
composer.addPass(smaaPass);
```

← 6. gltf后处理颜色异常(伽马校正)　　1. 射线Ray→