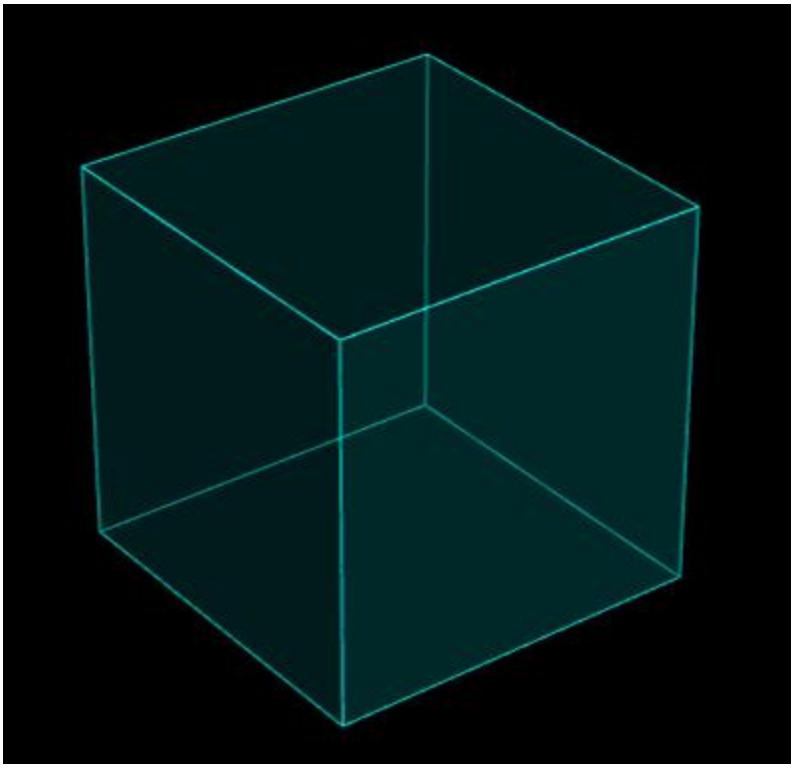
郭隆邦 2023-02-02

17. 模型边界线EdgesGeometry

借助 `EdgesGeometry` 可以给模型设置一个模型边界线。



长方体边线

先用 `EdgesGeometry` 重新计算长方体几何体，返回一个新的几何体，然后用线模型 `LineSegments` 模型渲染新的几何体即可。

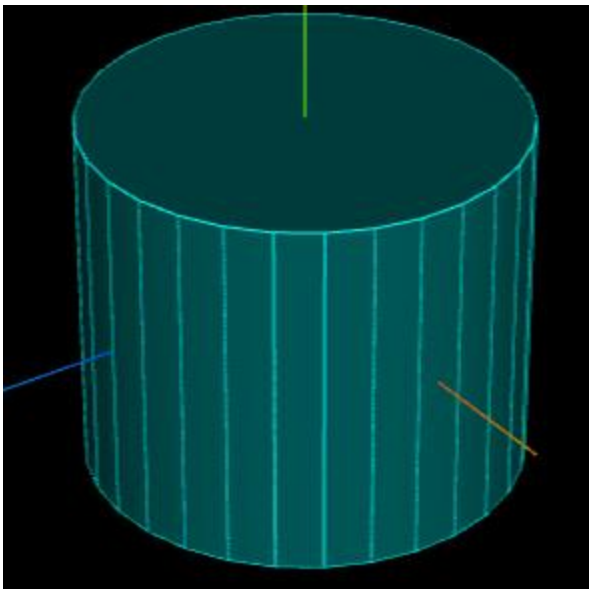
```js
const geometry = new THREE.BoxGeometry(50, 50, 50);
const material = new THREE.MeshLambertMaterial({
    color: 0x004444,
    transparent:true,
    opacity:0.5,
});
const mesh = new THREE.Mesh(geometry, material);

// 长方体作为EdgesGeometry参数创建一个新的几何体
const edges = new THREE.EdgesGeometry(geometry);
const edgesMaterial = new THREE.LineBasicMaterial({
```

```
  color: 0x00ffff,
})
const line = new THREE.LineSegments(edges, edgesMaterial);
mesh.add(line);
```

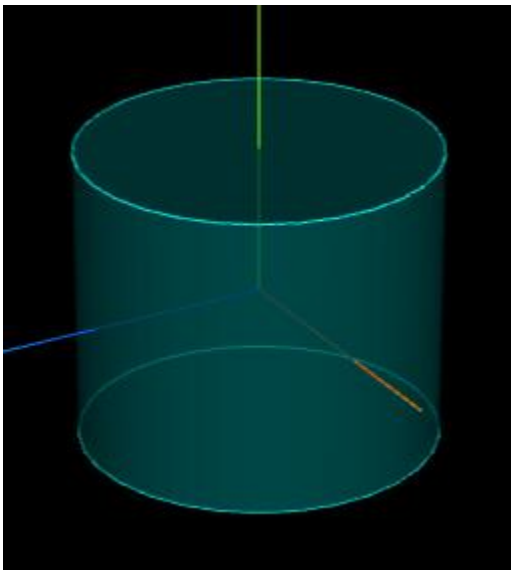
圆柱边线

```js
const geometry = new THREE.CylinderGeometry(60, 60, 100, 30);
const edges = new THREE.EdgesGeometry(geometry);
```

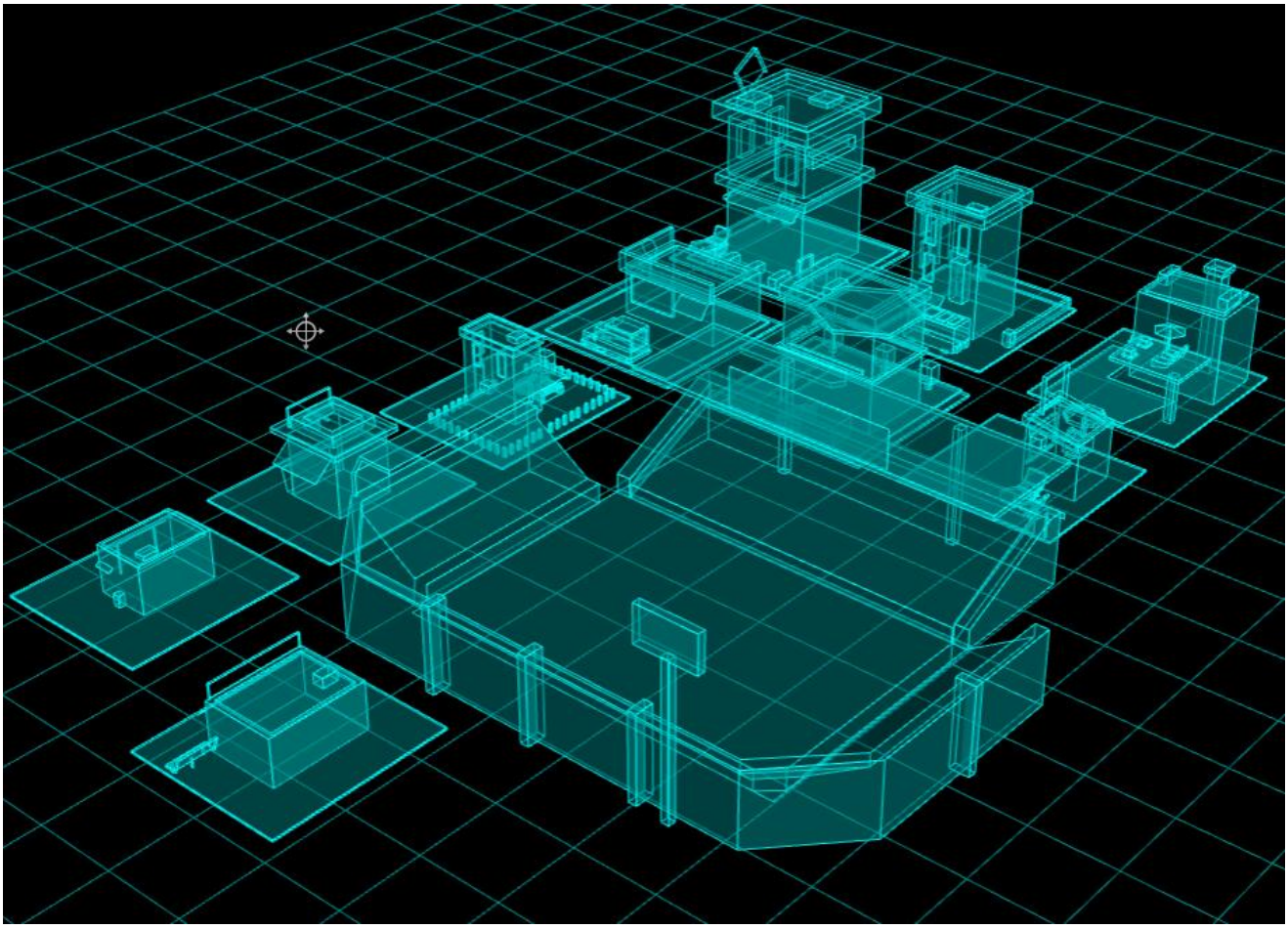


相邻面法线夹角大于30度，才会显示线条

```js
const edges = new THREE.EdgesGeometry(geometry,30);
```



外部gltf模型设置材质和边线



```js
loader.load("../建筑模型.gltf", function (gltf) {
    // 递归遍历设置每个模型的材质，同时设置每个模型的边线
    gltf.scene.traverse(function (obj) {
        if (obj.isMesh) {
            // 模型材质重新设置
            obj.material = new THREE.MeshLambertMaterial({
                color: 0x004444,
                transparent: true,
                opacity: 0.5,
            });
            // 模型边线设置
            const edges = new THREE.EdgesGeometry(obj.geometry);
            const edgesMaterial = new THREE.LineBasicMaterial({
                color: 0x00ffff,
            })
            const line = new THREE.LineSegments(edges, edgesMaterial);
            obj.add(line);
        }
```

```
    });
    model.add(gltf.scene);
})
```

← 16. 多边形Shape(内孔.holes)　　18. 几何体顶点颜色数数据→