

## 🎨 4. 环境贴图2

接着上节课的环境贴图给大家讲解。

### 环境贴图作用测试

实际生活中光源照射到一个物体上，这个物体反射出去的光线也会影响其他的物体，环境贴图就是用一种简单方式，近似模拟一个物体周边环境对物体表面的影响。

测试：对于PBR材质，如果threejs三维场景不添加任何光源，物体就是完全黑色的，你可以不添加任何光源，尝试只使用环境贴图，你会发现物体表面的颜色也能看到，这说明环境贴图其实相当于提供了物体周围环境发射或反射的光线。

测试：更换不同明暗的环境贴图，你会发现场景中模型的明暗也有变化。

### 场景环境属性 `.environment`

网格模型可以通过材质的 `.envMap` 属性设置环境贴图，如果一个glTF模型中所有的Mesh都要设置环境贴图就需要递归遍历glTF模型，给里面每个Mesh的材质设置 `.envMap`。

```
loader.load("../工厂.glb", function (glTF) {  
    // 递归遍历批量设置环境贴图  
    glTF.scene.traverse(function (obj) {  
        if (obj.isMesh) { //判断是否是网格模型  
            obj.material.envMap = textureCube; //设置环境贴图  
        }  
    });  
});
```

如果你希望环境贴图影响场景中scene所有Mesh，可以通过Scene的场景环境属性 `.environment` 实现,把环境贴图对应纹理对象设置为 `.environment` 的属性值即可。

```
// 环境贴图纹理对象textureCube作为.environment属性值,影响所有模型  
scene.environment = textureCube;
```

## 环境贴图色彩空间编码 `.encoding`

```
//如果renderer.outputEncoding=THREE.sRGBEncoding;环境贴图需要保持一致  
textureCube.encoding = THREE.sRGBEncoding;
```

---

← [3. 环境贴图.envMap\(金属效果\)](#)

[5. MeshPhysicalMaterial清漆层](#) →