

## 🔗 9. 相机控件OrbitControls

[视频讲解](#) 

平时开发调试代码，或者展示模型的时候，可以通过相机控件OrbitControls实现旋转缩放预览效果。

### OrbitControls使用

你可以打开课件案例源码测试下效果。

- 旋转：拖动鼠标左键
- 缩放：滚动鼠标中键
- 平移：拖动鼠标右键

### 引入扩展库OrbitControls.js

```
// 引入轨道控制器扩展库OrbitControls.js
import { OrbitControls } from 'three/addons/controls/OrbitControls.js';
```

注意：如果你在原生.html文件中，使用上面引入方式 `import { OrbitControls } from 'three/addons/controls/OrbitControls.js';`，注意通过 `<script type="importmap">` 配置。

```
<script type="importmap">
  {
    "imports": {
      "three": "../three.js/build/three.module.js",
      "three/addons/": "../three.js/examples/jsm/"
    }
  }
</script>
```

html

## 使用OrbitControls

```
// 设置相机控件轨道控制器OrbitControls
const controls = new OrbitControls(camera, renderer.domElement);
// 如果OrbitControls改变了相机参数，重新调用渲染器渲染三维场景
controls.addEventListener('change', function () {
    renderer.render(scene, camera); //执行渲染操作
}); //监听鼠标、键盘事件
```

js

## OrbitControls本质

OrbitControls本质上就是改变相机的参数，比如相机的位置属性，改变相机位置也可以改变相机拍照场景中模型的角度，实现模型的360度旋转预览效果，改变透视投影相机距离模型的距离，就可以改变相机能看到的视野范围。

```
controls.addEventListener('change', function () {
    // 浏览器控制台查看相机位置变化
    console.log('camera.position', camera.position);
});
```

js

---

← 8. 光源对物体表面影响

10. 平行光与环境光 →