

## 🎨 6. 模型或标签淡入淡出

如果你想实现一个模型或标签的淡入淡出效果，可以同过tweenjs控制模型或标签的透明度轻松实现。

### Tweenjs回调函数

twweenjs库提供了 `onStart` 、 `onUpdate` 、 `onComplete` 等用于控制动画执行的回调函数。

- `onStart` : 动画开始执行触发
- `onUpdate` : 动画执行过程中，一直被调用执行
- `onComplete` : 动画正常执行完触发

`.onUpdate(function(obj){})` 结构中，obj对应的是 `new TWEEN.Tween(pos)` 的参数对象pos。

```
const tween = new TWEEN.Tween(pos).to({x: 0}, 4000)
// 开始执行：动画片段tween开始执行的时候触发onStart
.tween.onStart(function(obj){
  ...
})
```

### 模型淡入

```
// 模型淡入
material.transparent = true; //开启透明计算
material.opacity = 0.0; //完全透明

// new TWEEN.Tween(material)
new TWEEN.Tween({opacity:material.opacity})
  .to({opacity:1.0}, 3000)
  .onUpdate(function(obj){
    material.opacity = obj.opacity
  })
  .onComplete(function(){
```

```
//动画结束：关闭允许透明，恢复到模型原来状态
material.transparent = false;
})
.start();
```

## 模型淡出

```
// 模型淡出
// new TWEEN.Tween(material)
new TWEEN.Tween({opacity:material.opacity})
.to({opacity:0.0}, 3000)
.onStart(function(){
    //动画开始：允许透明opacity属性才能生效
    material.transparent = true;
})
.onUpdate(function(obj){
    material.opacity = obj.opacity
})
.start();
```

## 模型HTML标签淡入淡出

three.js三维场景中的HTML元素标签淡入淡出和mesh对象淡入淡出类似操作。

单击模型弹出的标签淡入，而不是突然出现。

```
// HTML元素标签
const divTag = document.getElementById('tag');
```

```
const intersects = raycaster.intersectObjects(cunchu.children);
console.log('intersects', intersects);
if (intersects.length > 0) {
    // 通过.ancestors属性判断那个模型对象被选中了
    outlinePass.selectedObjects = [intersects[0].object.ancestors];
    //tag会标注在intersects[0].object.ancestors模型的局部坐标系原点位置
    intersects[0].object.ancestors.add(tag);
    span.innerHTML = intersects[0].object.ancestors.name; //修改标签数据

    // 标签淡出
    new TWEEN.Tween({opacity:0})
        .to({opacity:1.0}, 1000)
```

```

        .onUpdate(function(obj){
            //动态更新div元素透明度
            divTag.style.opacity = obj.opacity;
        })
        .start();
    }

```

原来代码，单击按钮，直接关闭标签。

```

// 鼠标单击按钮，关闭HTML标签
document.getElementById('close').addEventListener('click', function () {
    if (chooseObj) { //把原来选中模型对应的标签和发光描边隐藏
        outlinePass.selectedObjects = []; //无发光描边
        chooseObj.remove(tag); //从场景移除
    }
})

```

关闭模型标签淡出，逐渐消失。

```

// 鼠标单击按钮，关闭HTML标签
document.getElementById('close').addEventListener('click', function () {
    if (chooseObj) { //把原来选中模型对应的标签和发光描边隐藏
        outlinePass.selectedObjects = []; //无发光描边
        // chooseObj.remove(tag); //从场景移除
        new TWEEN.Tween({opacity: 1})
            .to({opacity: 0}, 400)
            .onUpdate(function (obj) {
                //动态更新div元素透明度
                divTag.style.opacity = obj.opacity;
            })
            .onComplete(function(){
                // 动画结束再从场景中移除标签
                chooseObj.remove(tag); //从场景移除
            })
            .start();
    }
})

```

