

鼠标无限移动 JS API Pointer Lock 简介

这篇文章发布于 2017年10月18日, 星期三, 01:16, 归类于 [JS API](#)。阅读 15778 次, 今日 5 次 [16 条评论](#)

by zhangxinxu from <http://www.zhangxinxu.com/wordpress/?p=6473>

本文可全文转载, 但需得到原作者书面许可, 同时保留原作者和出处, 摘要引流则随意。

一、Pointer Lock API是干嘛用的?

用一句话说明Pointer Lock API的作用就是: Pointer Lock API可以让你的鼠标无限移动, 脱离浏览器窗体的限制!

这对于一些需要鼠标控制的应用非常有用。举个例子: 某3D VR女友的游戏, 你鼠标向上移动, 则你的视角就会往下。



在默认网页环境下, 你鼠标移动到浏览器上边缘就不能再往上了, 结果, 本来还差一点点角度就能看到胖次, 结果卡在了关键地方, 岂不非常影响游戏体验。

但是如果使用了Pointer Lock API, 就不会出现这样的问题, 鼠标可以无限移动, 坐标可以无限增大, 完全脱离了浏览器窗口的限制, 可以完美享受沉浸式操作体验。

二、Pointer Lock API有哪些具体API名称?

目前, Pointer Lock API共支持3个属性, 2个方法和2个事件, 分别如下:

3个属性

`Document.pointerLockElement`

`Document.onpointerlockchange`

`Document.onpointerlockerror`

2个方法

`Element.requestPointerLock()`

`Document.exitPointerLock()`

2个事件

`pointerlockchange`

`pointerlockerror`

其中, 2个事件和其中2个属性是一一对应的, 因此, 我们实际上需要了解的知识点是下面这些:

`Document.pointerLockElement`，以及 `Element.requestPointerLock()`，`Document.exitPointerLock()` 以及 `pointerlockchange` 和 `pointerlockerror` 事件。

1. Document.pointerLockElement

指当前页面触发鼠标无限滚动的元素，通常使用语法为：

```
var element = document.pointerLockElement;
```

返回的是一个DOM元素对象，如果当前页面是非鼠标锁定状态，则返回值是 `null`。

2. Element.requestPointerLock()

可以让页面进入鼠标锁定状态（鼠标直接消失），鼠标无限滚动，坐标无限变化。通常使用语法为：

```
var element.requestPointerLock();
```

3. Document.exitPointerLock()

让页面从鼠标锁定状态退出，通常使用语法为：

```
document.exitPointerLock();
```

浏览器默认支持按下ESC键退出鼠标锁定状态，但是用户有时候更习惯于点击取消等，此时就可以使用 `document.exitPointerLock()` 进行设置。

4. pointerlockchange事件

当页面鼠标锁定状态改变的时候触发。例如：

```
document.addEventListener('pointerlockchange', function () {  
    // ...  
}, false);
```

5. pointerlockerror事件

当页面鼠标锁定失败的时候触发。例如当你试图同时锁定同一个页面的多个 `<iframe>` 时候，就会触发这个出错事件。

二、Pointer Lock API的图片无限3D旋转JS实例一则

看完了干巴巴的语法说明，我们还是来看一点鲜活的案例吧。

您可以狠狠的点击这里：[JS Pointer Lock与图片无限3D旋转demo](#)

默认进去会看到一个相貌平平的女生，然后点击这张图片：



此时鼠标会立即消失，同时图片就会根据你的鼠标位置开始翻江倒海，就像这样子：



而且鼠标的活动范围似乎没有边界，即使移动了一万个屏幕的宽度，我们的图片依然翻转个不停。这就是Pointer Lock API的特性表现。

相关代码如下：

```

```

JS部分：

```
var eleImage = document.getElementById('image');
if (eleImage) {
    // 起始值
    var moveX = 0, moveY = 0;
    // 图片无限变换的方法
    var rotate3D = function (event) {
        moveX = moveX + event.movementX;
        moveY = moveY + event.movementY;

        eleImage.style.transform = 'rotateX(' + moveY + 'deg) rotateY(' + moveX + 'deg)';
    };

    // 触发鼠标锁定
    eleImage.addEventListener('click', function () {
        eleImage.requestPointerLock();
    });

    // 再次点击页面，取消鼠标锁定处理
    document.addEventListener('click', function () {
        if (document.pointerLockElement == eleImage) {
            document.exitPointerLock();
        }
    });

    // 检测鼠标锁定状态变化
    document.addEventListener('pointerlockchange', function () {
        if (document.pointerLockElement == eleImage) {
            document.addEventListener("mousemove", rotate3D, false);
        } else {
            document.removeEventListener("mousemove", rotate3D, false);
        }
    }, false);
}
```

其中，应用了除了 `pointerlockerror` 事件外的其他所有属性和方法（见红色高亮）。

原理大致如下：

点击图片，执行 `eleImage.requestPointerLock()` 让页面进入鼠标锁定状态，此时会触发 `'pointerlockchange'` 事件，于是，给页面绑定鼠标移动改变图片旋转的方法，为了避免重复绑定，我们借助 `document.pointerLockElement` 判断页面的锁定状态。最后，为了方便取消页面的锁定状态，我们在页面上绑定了点击事件，通过 `document.exitPointerLock()` 取消页面的锁定状态。

需要说明的：

`event.movementX` 和 `event.movementY` 表示每次 `mousemove` 事件触发时候，距离上次移动的水平位置和垂直位置大小，而不是具体的某个坐标值。因此，需要和初始坐标不断的累加确定当前移动位置。

四、结束语

最后说下Pointer Lock API的兼容性。

由于Pointer Lock API是与鼠标相关的API，因此所有移动端都不支持，因为没有必要支持。

对于桌面浏览器，Chrome，Firefox以及Edge浏览器都是支持的，并且现在使用可以不加私有前缀，直接走起。IE并不支持，但这并不妨碍我们进行增强使用Pointer Lock API。

以上~

《CSS世界》签名版独家发售，包邮，可指定寄语，点击显示购买码

（本篇完） // 想要打赏？点击[这里](#)。有话要说？点击[这里](#)。



« [before\(\),after\(\),prepend\(\),append\(\)等新DOM方法简介](#)

[Chrome opacity非1时border-radius圆角边框剪裁问题](#) »

猜你喜欢

- CSS3 pointer-events:none应用举例及扩展
- 小卖弄：纯CSS实现的outline切换transition动画效果
- pointer-events:none提高页面滚动时候的绘制性能？
- jQuery - 鼠标经过(hover)事件的延时处理
- MooTools和jQuery库继承实现class切换对比
- js页面文字选中后分享到新浪微博实现
- HTML5终极备忘大全（图片版+文字版）
- CSSOM视图模式(CSSOM View Module)相关整理
- JS滚轮事件(mousewheel/DOMMouseScroll)了解
- jQuery boxy弹出层对话框插件中文演示及讲解
- 浏览器地理位置(Geolocation)API 简介

分享到：[+](#) [QQ](#) [微信](#) [微博](#) [贴吧](#) [豆瓣](#) [知乎](#) [收藏](#) [0](#)

标签：[API](#), [exitPointerLock](#), [pointer-events](#), [pointerlockchange](#), [pointerLockElement](#), [requestPointerLock](#), [鼠标事件](#)

发表评论（目前16条评论）

名称 (必须)

邮件地址(不会被公开) (必须)

网站

提交评论

1. nikolausliu说道：

2018年06月13日 11:55

结果发现鼠标移过浏览器编辑后，胖次部分是马赛克或者还没有构建...

[回复](#)



2. Stephen说道：

2017年12月12日 09:45

大神略略。你这网站速度真的很快啊，几乎所有页面都是毫秒开。请问你是怎么做到的？能不能专门写篇网站加载速度优化的文章？谢谢略略。

[回复](#)



木易S说道：

2018年09月13日 10:07

你用个不太差的服务器基本都不会慢的

[回复](#)



3. why说道：

2017年11月6日 10:17

相貌平平? ? ?

[回复](#)



阿萨德说道：

2017年11月8日 15:58

2313

[回复](#)



4. vivaxy说道：

2017年10月26日 15:55

var element.requestPointerLock();



这是 ES8 的新语法吗?

[回复](#)

5. 小媒体说道:

2017年10月25日 09:52

你发了，公司要上市。

[回复](#)



6. 上官说道:

2017年10月20日 09:37

相貌平平? ? ? 不存在的

[回复](#)



7. 幻白说道:

2017年10月19日 14:29

每天学一点，天天更牛逼

[回复](#)



8. 离殇说道:

2017年10月19日 14:29

涨见识了

[回复](#)



9. zzgzzg00说道:

2017年10月19日 11:33

一看举的例子这么污 就知道是你了

[回复](#)



10. boss说道:

2017年10月19日 11:19

safari下demo跑不了

[回复](#)



11. 书童说道:

2017年10月19日 09:43

不错

[回复](#)



12. 零谷说道:

2017年10月18日 16:26

又get到一个新技能

[回复](#)



13. 占楼说道:

2017年10月18日 14:21

占楼用的哈哈



回复

14. Real说道:

2017年10月18日 13:54

沙发~~ 写得很好, 又涨见识了

回复



最新文章

- » 常见的CSS图形绘制合集
- » 粉丝群第1期CSS小测点评与答疑
- » 分享三个纯CSS实现26个英文字母的案例
- » 小tips: 纯CSS实现打字动画效果
- » CSS/CSS3 box-decoration-break属性简介
- » CSS :placeholder-shown伪类实现Material Design占位符交互效果
- » 从天猫某活动视频不必要的3次请求说起
- » CSS vector-effect与SVG stroke描边缩放
- » CSS ::backdrop伪元素是干嘛用的?
- » 周知: CSS -webkit-伪元素选择器不再导致整行无效

今日热门

- » 常见的CSS图形绘制合集 (190)
- » 未来必热: SVG Sprite技术介绍 (119)
- » 粉丝群第1期CSS小测点评与答疑 (115)
- » HTML5终极备忘大全 (图片版+文字版) (93)
- » 让所有浏览器支持HTML5 video视频标签 (86)
- » Selectivizr-让IE6~8支持CSS3伪类和属性选择器 (82)
- » CSS3下的147个颜色名称及对应颜色值 (79)
- » 视区相关单位vw, vh..简介以及可实际应用场景 (76)
- » 写给自己看的display: flex布局教程 (76)
- » 小tips: 纯CSS实现打字动画效果 (76)



今年热议

- » 《CSS世界》女主角诚寻靠谱一起奋斗之人 (76)
- » 不借助Echarts等图形框架原生JS快速实现折线图效果 (64)
- » 看, for..in和for..of在那里吵架! (60)
- » 是时候好好安利下LuLu UI框架了! (47)
- » 原来浏览器原生支持JS Base64编码解码 (35)
- » 妙法攻略: 渐变虚框及边框滚动动画的纯CSS实现 (33)
- » 炫酷H5中序列图片视频化播放的高性能实现 (31)
- » CSS scroll-behavior和JS scrollIntoView让页面滚动平滑 (30)
- » windows系统下批量删除OS X系统.DS_Store文件 (26)
- » 写给自己看的display: flex布局教程 (26)

猜你喜欢

- CSS3 pointer-events:none应用举例及扩展

- 小卖弄：纯CSS实现的outline切换transition动画效果
- pointer-events:none提高页面滚动时候的绘制性能?
- jQuery - 鼠标经过(hover)事件的延时处理
- MooTools和jQuery库继承实现class切换对比
- js页面文字选中后分享到新浪微博实现
- HTML5终极备忘大全（图片版+文字版）
- CSSOM视图模式(CSSOM View Module)相关整理
- JS滚轮事件(mousewheel/DOMMouseScroll)了解
- jQuery boxy弹出层对话框插件中文演示及讲解
- 浏览器地理位置(Geolocation)API 简介