



## 求两直线的交点

### 直线与直线的位置关系

平面上的两条直线如果不平行, 那么他们一定相交, 并且有唯一的交点

### $Ax + By + C = 0$

直线一般式适用平面上任意直线

### 根据两点求解一般式的系数

设两个点为  $(x_1, y_1)$  ,  $(x_2, y_2)$ , 则有:  
 $A = y_2 - y_1$   
 $B = x_1 - x_2$   
 $C = x_2y_1 - x_1y_2$

### 直线标准式求系数

$Ax + By = C$

$A = y_2 - y_1$   
 $B = x_1 - x_2$   
 $C = Ax_1 + By_1$

### 直线一般式求交点

首先设交点坐标为  $(x, y)$ , 两线段对应直线的一般式为:

$a_1x + b_1y + c_1 = 0$   
 $a_2x + b_2y + c_2 = 0$

那么对 1 式乘  $a_2$ , 对 2 式乘  $a_1$  得:  
 $a_2 \cdot a_1x + a_2 \cdot b_1y + a_2 \cdot c_1 = 0$   
 $a_1 \cdot a_2x + a_1 \cdot b_2y + a_1 \cdot c_2 = 0$   
 两式相减得:

### 公告

昵称: pluscat  
 园龄: 3年4个月  
 粉丝: 6  
 关注: 9  
[+加关注](#)



2022年3月						
日	一	二	三	四	五	六
27	28	1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31	1	2
3	4	5	6	7	8	9

### 搜索

### 我的标签

$y = (c1 * a2 - c2 * a1) / (a1 * b2 - a2 * b1)$

同样可以推得：

$x = (c2 * b1 - c1 * b2) / (a1 * b2 - a2 * b1)$

如果 (x, y) 在两线段上, 则 (x, y) 即为答案, 否则交点不存在。

## 直线标准式求交点

首先设交点坐标为(x, y), 两线段对应直线的标准式为

$A1x + B1y = C1$

$A2x + B2y = C2$

将1式成以B2, 将2式乘以B1在相减

$A1B2x + B1B2y = B2C1$

$- A2B1x + B1B2y = B1C2$

$x = (B2C1 - B1C2) / (A1B2 - A2B1)$

同理可得

$y = (A1C2 - A2C1) / (A1B2 - A2B1)$

## 判断线段是否平行

如果两直线平行, 则有  $A1/B1 == A2/B2$ 。  
为了避免除零的问题, 可转化为  $A1*B2 == A2*B1$

## 利用一般式求两直线的交点

```
function lineIntersect(p0, p1, p2, p3) {
    var A1 = p1.y - p0.y,
        B1 = p0.x - p1.x,
        C1 = A1 * p0.x + B1 * p0.y,
        A2 = p3.y - p2.y,
        B2 = p2.x - p3.x,
        C2 = A2 * p2.x + B2 * p2.y,
        denominator = A1 * B2 - A2 * B1;

    return {
        x: (B2 * C1 - B1 * C2) / denominator,
        y: (A1 * C2 - A2 * C1) / denominator
    }
}
```

```
window.onload = function() {
    var canvas = document.getElementById("canvas"),
        context = canvas.getContext("2d"),
        width = canvas.width = window.innerWidth,
        height = canvas.height = window.innerHeight;

    var p0 = {
        x: 100,
        y: 100
    },
    p1 = {
        x: 500,
        y: 500
    },
```

linux(2)

CDN(1)

### 随笔分类

git(1)

javascript(48)

linux(5)

linux系统编程(3)

nodeJs(2)

python(3)

react(6)

vim(1)

webpack(10)

web应用开发实践(1)

操作系统(10)

概率(4)

计算机网络(8)

数据结构与算法(15)

线性代数(6)

### 随笔档案

2021年7月(2)

2021年6月(4)

2020年11月(4)

2020年10月(7)

```
p 2 = , {
    x: 600,
    y: 50
},
p 3 = {
    x: 80,
    y: 600
};

context.beginPath();
context.moveTo(p0.x, p0.y);
context.lineTo(p1.x, p1.y);
context.moveTo(p2.x, p2.y);
context.lineTo(p3.x, p3.y);
context.stroke();

var intersect = lineIntersect(p0, p1, p2, p3);

context.beginPath();
context.arc(intersect.x, intersect.y, 20, 0, Math.PI * 2, false);
context.stroke();

function lineIntersect(p0, p1, p2, p3) {
    var A1 = p1.y - p0.y,
        B1 = p0.x - p1.x,
        C1 = A1 * p0.x + B1 * p0.y,
        A2 = p3.y - p2.y,
        B2 = p2.x - p3.x,
        C2 = A2 * p2.x + B2 * p2.y,
        denominator = A1 * B2 - A2 * B1;

    return {
        x: (B2 * C1 - B1 * C2) / denominator,
        y: (A1 * C2 - A2 * C1) / denominator
    }
};
```

## 判断直线平行和相交的情况

交点在一条直线的延长线上或者交点在两条直线的延长线上不画出交点

```
function segmentIntersect(p0, p1, p2, p3) {
    var A1 = p1.y - p0.y,
        B1 = p0.x - p1.x,
        C1 = A1 * p0.x + B1 * p0.y,
        A2 = p3.y - p2.y,
        B2 = p2.x - p3.x,
        C2 = A2 * p2.x + B2 * p2.y,
        denominator = A1 * B2 - A2 * B1;

    // 如果分母为0 则平行或共线，不相交
    if(denominator == 0) {
        return null;
    }

    var intersectX = (B2 * C1 - B1 * C2) / denominator,
        intersectY = (A1 * C2 - A2 * C1) / denominator,
        rx0 = (intersectX - p0.x) / (p1.x - p0.x),
        ry0 = (intersectY - p0.y) / (p1.y - p0.y),
        rx1 = (intersectX - p2.x) / (p3.x - p2.x),
        ry1 = (intersectY - p2.y) / (p3.y - p2.y);

    /** 2 判断交点是否在两条线段上 **/
```

2020年9月(4)

2020年8月(1)

2020年6月(1)

2020年2月(1)

2019年12月(1)

2019年11月(1)

2019年10月(5)

2019年9月(2)

2019年8月(1)

2019年7月(2)

2019年5月(7)

更多

### 阅读排行榜

1. scp命令的使用(16652)

2. openssl命令使用(11179)

3. 求两直线的交点(7771)

4. RegExp使用技巧(7628)

5. 传输层协议总结(6615)

### 评论排行榜

1. vim-plugin(12)

### 推荐排行榜

1. js函数式编程(1)

```
        if (
            // 交点在线段1上
            ((rx0 >= 0 && rx0 <= 1) || (ry0 >= 0 && ry0 <= 1)) &&
            // 且交点也在线段2上
            (( r x l  >= 0 && rxl <= 1) || (ryl >= 0 && ryl <= 1))) {
                return {
                    x: intersectX,
                    y: intersectY
                };
            }
            else {
                return null;
            }
        }
    }
}
```

分类: javascript

好文要顶 关注我 收藏该文



pluscat  
关注 - 9  
粉丝 - 6

+加关注

0  
推荐

0  
反对

« 上一篇: 微任务防抖

» 下一篇: 向量的运算

posted @ 2020-10-11 23:33 pluscat 阅读(7771) 评论(0) 编辑 收藏 举报

刷新评论 刷新页面 返回顶部

登录后才能查看或发表评论，立即 登录 或者 逛逛 博客园首页

编辑推荐：

- 突破限制，CSS font-variation 可变字体的魅力
- 浅谈 C# 字符串构建利器 StringBuilder
- 老项目的倔强 —— 性能优化篇
- 2021 .NET Conf China 主题分享之-轻松玩转.NET大规模版本升级
- 理解 OAuth2.0 协议和授权机制

最新新闻：

- 勒索老黄未果！黑客公布英伟达核心源代码，超40万个文件、75GB机密数据
- 不断加深的危机迫使物理学家重新思考自然法则的结构
- “拯救者”拯救不了联想手机
- 骨传导耳机不会影响听力，是真的咩？
- 苹果坚持职能制，组织转型还有必要吗？
- » 更多新闻...

2. scp命令的使用(1)

## 最新评论

1. Re:vim-plug

@pluscat 谢谢，已经修改好了...

--danshu00

2. Re:vim-plug

@danshu00 是在这个配置中~/vimrc用vim打开执行:PlugInstall，注意大小写...

--pluscat

3. Re:vim-plug

@pluscat 是的，我是输入后返回E492：not an editor command: plugininstall...

--danshu00

4. Re:vim-plug

@danshu00...

--pluscat

5. Re:vim-plug

@pluscat 这个可能性可以排除掉，因为我确实是从终端进入vim，然后再vim的命令行模式的下端输入的：plugininstall等命令的...

--danshu00