

分享一个即插即用的私藏缓动动画JS小算法

这篇文章发布于 2017年01月17日, 星期二, 00:56, 归类于 JS实例。 阅读 28804 次, 今日 24 次 [28 条评论](#)

by zhangxinxu from <http://www.zhangxinxu.com/wordpress/?p=5875>

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、动画算法的故事



人是一种恋旧的动物, 即使过了很多年, 即使小屁孩已经学会了打酱油, 但是, 这心中那, 总是时不时会想起初恋的那个她, 虽未提及, 但在心里。

如果动画算法是恋人的话, 那也有类似的故事。

当年还是个青涩男孩的时候, 什么都是懵懵懂懂, 实现效果要么干巴巴, 要么拿来主义。直到一次偶然的邂逅, 一次冥冥中注定的邂逅, 我遇到了她, 她是如此简单可爱, 让我过目不忘。于是, 一下子自己仿佛打开了一扇新的窗, 有了她的相伴, 实现的效果不再是干巴巴的, 而是开始有了绚丽的色彩, 开始有了灵动的气息。

后来, 我成长了, 接触了越来越多的外面的世界, 遇到了越来越多是动画算法, 这些算法确实很美, 实现的效果更佳的炫丽, 甚至将她们全部收纳整理成了后宫家族(可参考之前[“如何使用Tween.js各类原生动画运动缓动算法”](#)这篇文章), 但是, 实际开发的时候, 使用的最多的还是最初的那个她, 不仅是因为她给我带来了许多美好的回忆, 更重要的是这么多年过去了, 她一直保持着那份简单, 和外面的那些妖艳*货完全不一样。

而这个相伴自己最久, 最无法忘怀的、使用最多的她就是本文要分享的私藏的动画小算法。

[Tween.js](#)中的[动画算法](#)虽然标准全面且强大, 但是, 里面的公式略复杂, 根本就记不住, 每次使用都要去找一下, 甚是麻烦。

而且, 平时开发, 绝大多数场景下, 我们并不需要是那种弹来弹去的浮夸效果, 有点缓动就可以, 本文要分享的动画小算法就是一个简单的缓动小算法。

二、即插即用的缓动小算法

原理如下:

假设要从数值A变化到数值B, 如果是线性运动, 则每次移动距离是一样; 如果是缓动, 每次移动距离不一样。那如何才能不一样呢? 很简单, 按比例移动就可以。

例如: 每次移动剩余距离的一半。

对吧, 超容易理解的。

比方说: 你和初恋之间距离是64, 每秒移动一半, 则, 你们之间的距离下一秒就是32, 再下一秒就是16, 然后8, 然后4, 然后2, 然后1, 然后.....你们就在一起了。你们在一起的这个过程就是一个典型的先快后慢的缓动运动过程, 如下示意图:



用一个简单的公式表示就是:

$$A = A + (B - A) / 2$$

翻译一下就是:

$$\text{我下一秒的位置} = \text{现在位置} + \text{现在和初恋之间距离的一半}$$

是不是很好理解。

而上面的 $A = A + (B - A) / 2$ 就是本文要介绍的即插即用的缓动小算法。

当然要实际使用还是需要做一点点的处理的, 首先, 既然是运动, 那就离不开定时器, 我们可以使用 `requestAnimationFrame`, 对于不支持的浏览器, 可以使用下面的兼容代码:

```
// requestAnimationFrame的兼容处理
if (!window.requestAnimationFrame) {
  requestAnimationFrame = function(fn) {
    setTimeout(fn, 17);
  };
}
```

下面我们套用一个简单的实例，看看这个算法是如何应用的，就是非常常见的返回顶部效果。很多网站返回顶部要么直接“啪”到顶部，要么动画按部就班没有灵性，这里演示下如何使用寥寥数行代码，实现带缓动的返回顶部效果。点击下面的按钮即可看到返回顶部效果：

点击按钮执行的是下面的 `backToTop()` 方法：

```
// 滚动到顶部缓动实现
// rate表示缓动速率，默认是2
var backToTop = function (rate) {
    var doc = document.body.scrollTop? document.body : document.documentElement;
    var scrollTop = doc.scrollTop;

    var top = function () {
        scrollTop = scrollTop + (0 - scrollTop) / (rate || 2);

        // 临界判断，终止动画
        if (scrollTop < 1) {
            doc.scrollTop = 0;
            return;
        }
        doc.scrollTop = scrollTop;
        // 动画gogogo!
        requestAnimationFrame(top);
    };
    top();
};
```

其中，代码的核心是：

```
scrollTop = scrollTop + (0 - scrollTop) / (rate || 2);
```

`scrollTop` 表示公式的A, 滚动到顶部滚动高度是 `0`，因此，上面的 `0`，实际上就是公式的B, 而公式中的 `2` 表示缓动速率，实际开发的时候是可以灵活调整的，缓动速率范围是 `1` 到无穷大，速率值越小，运动越快。比如说上面的返回顶部效果，我们把缓动速率改成 `4`，点击下面的按钮感受效果：

可以明显感觉到，返回顶部使用的时间变长了，速度变慢了，缓动体验也更明显了。

等比例靠近理论上最终只会无穷靠近，并不会真正的相等，也就是动画永远没有结束的时候，所以说需要做一个临界判断，也就是距离小到一定数目的时候，直接等于目标值，并终止动画。例如，上面的返回顶部，就是当距离顶部滚动高度小于 `1` 的时候，直接返回顶部，并终止动画。

```
if (scrollTop < 1) {
    doc.scrollTop = 0;
    return;
}
```

我们最近刚上线不久的新版起点移动站 (<http://m.qidian.com>) 的返回顶部效果就是使用的这个缓动小算法。



当然，起点M站使用这个动画小算法的地方不止这一个，比方说首页最近阅读的小球是可以移动的，惯性运动动画效果就是用的该算法，还是书籍阅读页面水平看书模式时候移来移去的翻页效果，也是此算法；垂直模式时候点击翻页也是该算法等等。

用得非常多，因为简单，不需要查资料直接可以写出来，只要记住“等比例靠近”这个词，就不会忘记该怎么实现，永远忘不了的初恋就是这种感觉。

三、即插即用的缓动小算法变身

如果项目很多地方使用该算法，每次都写一遍 `requestAnimationFrame` 和边界判断是很啰嗦的，于是，我们可以把算法变个身，例如下面这样：

```
Math.easeout = function (A, B, rate, callback) {
    if (A == B || typeof A != 'number') {
        return;
    }
    B = B || 0;
    rate = rate || 2;

    var step = function () {
        A = A + (B - A) / rate;

        if (A < 1) {
            callback(B, true);
            return;
        }
        callback(A, false);
        requestAnimationFrame(step);
    };
    step();
};
```

```
};
    step();
};
```

其中：

- **A** 是起始位置；
- **B** 是目标位置；
- **rate** 是缓动速率；
- **callback** 是变化的位置回调，支持两个参数，**value** 和 **isEnding**，表示当前的位置值（数值）以及是否动画结束了（布尔值）；

于是，我们的返回顶部效果可以这么使用：

```
var doc = document.body.scrollTop? document.body : document.documentElement;
Math.easeout(doc.scrollTop, 0, 4, function (value) {
    doc.scrollTop = value;
});
```

可以看到效果是一样的棒棒哒。

更关键点是我们的算法可以全局无限制重复利用啦！

四、结束语

本文有3个按钮，每个按钮点击都会有带缓动的返回顶部的效果，如果您发现您的浏览器按钮点击没效果，或者没按钮，则多半您访问的不是原出处。

根据我自己的猜测，**Tween.js**标准的缓动算法中，应该有类似本文的小算法的，应该就是某个easeOut方法。但是，具体实现上还是会有一些差异的，**Tween.js**是基于时间来做算法的，而本文的**Math.easeout**算法是基于缓动速率的，最终的时间是不固定的，受起始和终止位置的距离的影响，也正是这个原因，本文的缓动动画算法更好理解更容易记忆，因为现实世界距离是真实的容易感知的，时间是虚幻的无法捕捉的，正好和本文的算法距离驱动而非时间驱动相映射，容易形成共识。

恩，就说这么多吧。

感谢阅读，欢迎交流，更欢迎大力分享，你瞧，我把珍藏的初恋都分享出来了，还不点赞分享支持下。



《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](#)。



« [HTML5语音合成Speech Synthesis API简介](#)

[基于VoiceOver的移动web站无障碍访问实践](#) »

猜你喜欢

- [jQuery-火焰灯效果导航菜单](#)
- [CSS3动画那么强，requestAnimationFrame还有毛线用？](#)
- [如何使用Tween.js各类原生动画运动缓动算法](#)
- [开源移动端元素拖拽惯性弹动以及下拉加载两个JS](#)
- [JS检测PNG图片是否有透明背景、抠图等相关处理](#)
- [JS实现照片图片变成黑白线条线稿](#)
- [jquery之图片左右切换动画效果](#)
- [JavaScript实现新浪微博文字放大显示动画效果](#)
- [jQuery之使用slideToggle实现垂直下拉菜单](#)
- [CSS3 transition实现超酷图片墙动画效果](#)
- [我要上头条——鑫表情包1.0正式发布](#)

分享到：[微信](#) [QQ](#) [微博](#) [豆瓣](#) [知乎](#) [百度](#) [贴吧](#) [论坛](#) [博客](#) [邮箱](#) [打印](#) [收藏](#) [分享](#) [0](#)

标签：[easeout](#), [动画](#), [算法](#), [缓动](#)

发表评论（目前28条评论）

<input type="text"/>	名称 (必须)
<input type="text"/>	邮件地址(不会被公开) (必须)
<input type="text"/>	网站

提交评论

1. 琉璃说道:

2018年11月2日 16:09

诙谐幽默，甚是喜欢

[回复](#)



2. ddd说道:

2017年11月9日 08:27

ie都不好使啊 机上那个佳能处理的也不好啊

[回复](#)



测过说道:

2018年01月23日 11:04

我试过了可以的啊，你是不是使用了DOM2监听按钮点击事件啊？IE不支持DOM2级的事件

[回复](#)



3. 可儿说道:

2017年05月21日 14:02

旭哥我和我的姐们一直把你认为是神一样的存在，都超级的喜欢你。

[回复](#)



4. 徐徐徐徐说道:

2017年03月16日 14:43

+1s 心有感触

[回复](#)



5. 徐昆龙说道:

2017年03月1日 09:52

这个动画，一直没搞明白，看了这篇后，都懂了，真是好文！

[回复](#)



6. 依韵说道:

2017年02月22日 13:11

这个只能往上动 而且结束必须是0，我改写了下：

```
var _easeout = function(start, end, rate, callback) {
  var _end = end;
  if (start == end || typeof start != 'number') {
    return;
  }
  end = end || 0;
  rate = rate || 2;
```

```
var step = function() {
  start = start + (end - start) / rate;
  if (Math.abs(start - _end) < 1) {
```

```
// if (start < 1) {
  console.log('end');
  callback(end, true);
  return;
}
callback(start, false);
requestAnimationFrame(step);
};
step();
};
```

调用

```
var doc = document.body.scrollTop !== undefined ? document.body : document.documentElement;
console.log('start', doc.scrollTop);
console.log('end', end);
_easeout(doc.scrollTop, end, 2, function(value) {
  doc.scrollTop = value;
});
```

结束的判断修改了。还有

var doc = document.body.scrollTop !== undefined ? document.body : document.documentElement;
没有滚动 在顶部时，document.body.scrollTop为0 会取成 document.documentElement; 就不能滚动了。

[回复](#)



7. 菜鸟001说道:

2017年02月10日 14:59

最后封装的算法有点问题, if (A < 1) 改成 if(Math.abs(A-B) < 1),是否是这样?

[回复](#)



8.

t说道：
2017年02月4日 15:54

你好 那个requestAnimationFrame兼容里面 那个 setTimeout(fn,17) 这个17是随便写一个数字 还是有啥意义啊！

回复

张 鑫旭说道：
2017年02月7日 16:32

1000/60 ≈ 17

回复

t说道：
2017年02月8日 03:50

大神 我又百度了一下 我现在的理解是 默认60ms变化一次 1000ms动了17次 应该是这样了 谢谢！

回复

t说道：
2017年02月8日 04:38

我好像看错了 每秒60帧？ 这误差不多吧 还是拿来主义吧。

回复

n说道：
2017年02月8日 13:23

电脑显示屏那一帧刷新频率每秒（1000ms）60帧。所以1000/60约17ms一帧是屏幕显示的合理时间。。反正我感觉是这样的。。

回复

9.

t说道：
2017年02月4日 11:39

新年快乐！

回复

10.

ziven27说道：
2017年01月24日 20:22

张老师，
在Math.casout中，当B>A的时候函数会一直执行，不会停止；
我的解决办法是
把：
if (A < 1) {
callback(B, true);
return;
}
改成：
if (Math.abs(A - B) < 1) {
callback(B, true);
return;
}
回复

11.

我是来评论的说道：
2017年01月22日 17:03

关注点偏移~
初恋！

回复

12.

yyssjj33说道：
2017年01月21日 16:40

建议可以来个fibonacci序列的

回复

13.

勇敢的南瓜说道：
2017年01月21日 10:15

震精！

回复

14.

tians说道：
2017年01月20日 13:11

棒！

回复

15.

我就想知道说道：
2017年01月19日 13:05

大神,是怎么做到在自己博客网站写点击事件的

回复

张 鑫旭说道：
2017年01月20日 21:42

自己博客自己管理员当然可以啦~

回复

16.

阿牛说道：
2017年01月18日 15:41

应该还有退出动画的机制不然会一直执行

回复

17. 喵喵说道:

2017年01月17日 17:00

最后的按钮设绑执行的方法

回复



18. Simon说道:

2017年01月17日 16:58

原来玩大神的代码也并不都是大神级别的，看到这些小代码让人有一种返璞归真的感觉。新的一年，我也要加油！
btw:rate2和4在效果感觉上差别太小，只是慢了一倍的速率，是否需要用大一点的rate值来说明会更好？

回复



19. 海阔天空说道:

2017年01月17日 09:53

不错，谢谢

回复



20. Muffin说道:

2017年01月17日 09:08

按钮无法执行？

回复



张鑫旭说道:

2017年01月17日 10:24

移动端我兼容下~

回复

最新文章

- » 常见的CSS图形绘制合集 (119)
- » 粉丝群第1期CSS小测点评与答疑 (115)
- » 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 (119)
- » 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） (85)
- » 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》女主角诚挚邀请一起奋斗之人 (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

- » jQuery-火焰灯效果导航菜单
- » CSS3动画那么强，requestAnimationFrame还有毛线用？
- » 如何使用Tween.js各类原生动画运动缓动算法

- 开源移动端元素拖拽惯性弹动以及下拉加载两个JS
- JS检测PNG图片是否有透明背景、抠图等相关处理
- JS实现照片图片变成黑白线条线稿
- jquery之图片左右切换动画效果
- JavaScript实现新浪微博文字放大显示动画效果
- jquery之使用slideToggle实现垂直下拉菜单
- CSS3 transition实现超酷图片墙动画效果
- 我要上头条——鑫表情包1.0正式发布