

炫酷H5中序列图片视频化播放的高性能实现

这篇文章发布于 2018年05月1日, 星期二, 19:12, 归类于 [JS实例](#)。阅读 17123 次, 今日 11 次 [31 条评论](#)

by zhangxinxu from <http://www.zhangxinxu.com/wordpress/?p=7526>

本文可全文转载, 但需要保留原作者和出处, 摘要引流则随意。

一、爆款H5中的炫酷场景的技术实现

每年都会进出一些爆款H5, 这些H5通常会有一些酷酷的场景变换。

例如[网易Julia H5](#)。创意很棒, 传播效果很广, 其技术实现就相当简单粗暴, 直接一个3分多钟的不带停顿的视频:



类似使用视频的爆款H5还有很多, 确实是一个非常不错的解决方案。

但是, 视频的实现也有局限:

1. iOS下不能自动播放, 需要至少touch一次屏幕, 这个有时候还挺烦人的, 例如我们想做一个H5 app 闪屏的时候就蛋疼了。
2. 不能在中间穿插棒棒的交互效果, 例如, 需要视频某一帧暂停, 鼠标hover或者touch的时候, 当前画面有交互效果, 就很不好处理。
3. 播放的速率不能随心所欲控制, 视频完成也就定死了。
4. 如果有些信息是动态的, 需要与用户信息关联, 则视频方案也会面临很大的调整, 因为总不可能每一个用户生成一个不一样的视频, 需要辅助额外手段满足需求 (例如CSS覆盖定位)。

此时又当如何处理呢?

我们可以使用序列图片, 通过JS脚本, 来模拟视频播放效果, 以上所有局限将通通可以规避。

二、序列图片视频化技术实现方法很多

序列图片视频化技术实现方法很多，例如：

1. 合在一张大图上，使用CSS3 `animation` 控制 `background-position` 实现播放效果。此方法优点是方便快捷，但是，`background-position` 定位性能不怎么样，只适用于小元素小动画，例如一些loading效果，如果是全屏的大图切换，则在移动设备上可以明显感觉到卡顿；
2. 合在一张大图上，`transform` 定位。性能还是撑不住；
3. 图片一次性在页面上，依次控制显隐。性能有所提高，但如果图片序列上百帧，图片尺寸较大，性能还是撑不住，客户端说不定会直接闪退。
4. 页面上一个 `` 元素，然后不断改变 `src` 地址，不好意思，也不是很快。

总而言之，上面各种方法都是理论上可行，但实践下来的性能总不尽如人意。

那有没有什么高性能的实现方法呢？

有，一种是使用canvas绘图，但如果想要动态插入其他UI丰富的DOM结构（如登录模块），就比较麻烦；还有一种就是下面这个方法，实践下来性能可以，体验比较好，实现成本也不高。

三、序列图片视频化技术高性能实现方法

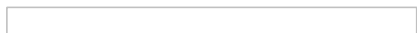
实现原理如下：

1. 图片DOM对象预加载，放在内存中；
2. 播放开始，页面append当前图片DOM，同时移除上一帧DOM图片（如果有），保证页面中仅有一个图片序列元素；

对，很简单，没什么高超的技巧，但就是这种实现策略，对页面的开销是上面几种方法中最小的，最终运行体验是最好的。

眼见为实，您可以狠狠地点击这里：[序列图片实现视频播放效果demo](#)

loading完毕，就可以看到一段播放效果——本人飞吻一枚：



效果之流畅，体验之良好，十有八九都会认为是视频，其实不是，就是图片，不断的图片DOM增删实现的类似视频效果。

核心JS代码如下（完整代码见demo），假设 `container` 是容器元素，我们的图片已经预加载到 `store` 对象中，结构如下：

```
var store = {
  length: 47,
  1: img1,
  2: img2,
  ...
  47: img47
};
```

则有：

```
var index = 1;
container.innerHTML = '';
```

```
// 依次append图片对象
var step = function () {
    if (store[index - 1]) {
        container.removeChild(store[index - 1]);
    }
    container.appendChild(store[index]);
    // 序列增加
    index++;
    // 如果不超过最大限制，播放下一帧
    if (index <= 47) {
        // 42是按照每秒24帧计算的值
        setTimeout(step, 42);
    }
};
step();
```

上面代码红色高亮是实现的关键，`container.removeChild(store[index - 1])` 移除之前一帧图片DOM，`container.appendChild(store[index])` 则是插入当前一帧DOM，人的肉眼习惯连续性感知事物，因此，这种删除和添加，用户是无感知的，于是一个流程的播放效果即达成，根据实践，就算每帧图片在几百K大小主流设备也能hold住。

由于本质上播放的是DOM对象，因此，我们不仅可以播放图片DOM，还可以是有着丰富HTML结构的<div>元素，于是，什么样的交互实现都不在话下，比方说视频中要出现用户的姓名，怎么办，很简单啊，<div>元素中定位下就好了。

现在，技术实现已经对设计没有任何限制啦，剩下的就是产品和设计的创意，下一个爆款H5就是你了！

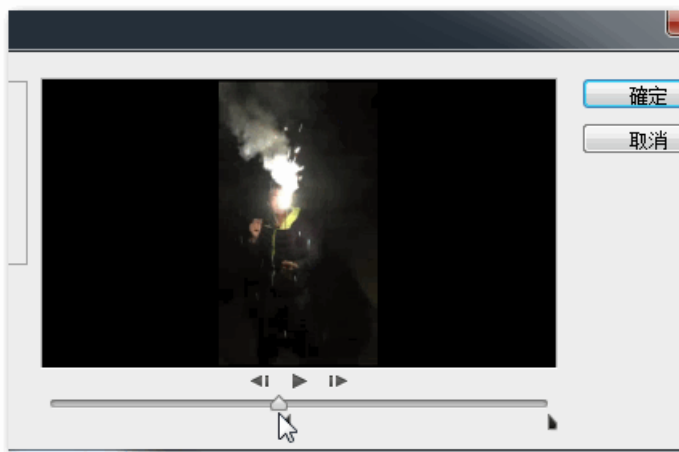
三、番外技能：如何把视频变成序列图片？

如果贵厂有设计师，让设计师处理，基本上，AE之类软件应该都了解。

如果没有设计师，要前端小伙伴自己处理，怎么办？

拿本文的序列图片举例，我是这么处理的：

1. 手机视频QQ传到自己电脑；
2. 打开Photoshop，然后：文件-导入-视频帧到图层，会出现一个窗口，可以剪切需要导入的视频范围；



如果导入视频很长，建议先剪辑，再导入，除非你的电脑是iMac Pro这种级别的。

3. 不做任何处理，直接：“文件-脚本-将图层导出到文件”；

4. node.js小工具批量重命名了下（导出图片名称后面序号是准确序号），工具使用的JavaScript代码如下：

```
var fs = require("fs");
fs.readdir('./', function(err, files) {
  files.forEach(function(filename) {
    if (/jpg$/.test(filename) == false) { return; }
    // 确定新旧文件名称
    var oldPath = './' + filename, newPath = './' + filename.split(' ')[1];
    // 重命名走起
    fs.rename(oldPath, newPath, function() {
      console.log(filename + '重命名成功!');
    })
  });
});
```

四、关于性能其他需要注意的

人眼的跟踪能力要比大猩猩之类要弱的，因此，实际开发，并不一定需要每秒24帧的播放速率，你每秒18帧，对于一个H5运营活动而言，用户是无感知的。每秒18帧的播放可以节约不少请求和加载数据量，性能上也能有所提高，权衡来看，是推荐的，毕竟我们不是去参加动画比赛，是一个在线的web产品。

设计师喜欢使用非常高清的图片，实际上，没有必要，注意度，2倍尺寸，30%~40%的图片质量足够了，效果也非常好，这也是经过实践的，大家如果和设计师意见不一致，就可以让她看我写的这段话。有效降低不必要的图片尺寸，可以大大节约内存的开销，也是可以提高播放的性能和品质的。

于是，三管齐下：高性能技术实现策略，适当降低帧率，优化图片尺寸，必定助你H5炫酷效果流畅至极，好评如潮，boss交口称赞！

以上就是本文内容，感谢阅读，如果文中有表述不准确的地方，欢迎指正！

《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](#)。



« JS检测PNG图片是否有透明背景、抠图等相关处理

纯前端实现可传图可字幕台词定制的GIF表情生成器 »

猜你喜欢

- 从天猫某活动视频不必要的3次请求说起
- canvas图形绘制之星空、噪点与烟雾效果
- 基于clip-path的任意元素的碎片拼接动效
- canvas实现iPhoneX炫彩壁纸屏保外加pixi.js流体动效
- CSS CSS3实现3D开门动画效果
- 回流与重绘：CSS性能让JavaScript变慢？
- 翻译 - CSS Sprites:实用技术还是生厌之物？

- 移动页面加速google的AMP和百度的MIP简介
- 让所有浏览器支持HTML5 video视频标签
- 揭秘视频网站video视频倍速播放的实现
- HTML5新增的form属性简介

分享到:  0

标签: HTML5, 优化, 动效, 序列图片, 性能, 视频

发表评论（目前31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. 课程说道:

2019年01月8日 18:37

请问声音怎么处理呢,

[回复](#)



2. 大东说道:

2018年09月17日 10:57

作为一个小白, 想请教一下大神, 视频短的可以用gif (2-3s), 长的广告图片又有太多 (比如15s), 这个方法的适用场景有哪些, 优势在哪。

[回复](#)



3. xxx说道:

2018年08月19日 23:08

示例页面在IE下报错。

SCRIPT438: 对象不支持“append”属性或方法

文件: image-sequence-frame-play.html, 行: 221, 列: 4

```
if (store[index - 1]) {  
  eleContainer.removeChild(store[index - 1]);  
}  
//HERE !!!  
eleContainer.append(store[index]);  
// 序列增加  
index++;
```

[回复](#)



张鑫旭说道：
2018年08月21日 14:37
感谢反馈，已修～
[回复](#)



4. **linkstar** 说道：

2018年07月17日 11:49

序列图片实现视频播放效果demo
这个点开都是undefined

[回复](#)



张鑫旭说道：
2018年07月17日 23:58
感谢反馈，已修复。
[回复](#)



5. **SilenceSea** 说道：

2018年05月30日 21:34

学习了

[回复](#)



6. **cocody** 说道：

2018年05月18日 16:30

张大神可以详细说说为什么增删 DOM 元素比切换 src 更快么？

[回复](#)



7. **读心** 说道：

2018年05月15日 20:33

挺好玩的呢！！！！

[回复](#)



8. **nicholasurey** 说道：

2018年05月7日 13:31

思路很棒

[回复](#)



9. **滴滴答答哪** 说道：

2018年05月7日 09:18

julia 为什么我先想到了一个女演员

[回复](#)



10. **Plus.Zhang** 说道：

2018年05月4日 17:53

对于有声音的视频来说似乎方案不是那么可行啊

[回复](#)



11.

沉江、小鱼说道：

2018年05月4日 15:32

拜读了大神的《CSS世界》，受益匪浅，写得很好，真的，糅合了自己的思考和想法，这是最难能可贵的，读书就像和作者交流，就是要把自己的想法和个性表达出来，写出来，不拘泥于规范和冷冰冰的教程，思想才能碰撞出火花嘛

回复

张鑫旭说道：

2018年05月4日 16:59

感谢支持！

回复


12.

firstBlood说道：

2018年05月4日 11:42

我一般是用canvas, 很多逻辑都会写到render里面, 主要是参考的threejs里面的一些做法吧

回复


13.

DH说道：

2018年05月4日 11:02

相见恨晚，感谢提供思路

回复


14.

码农说道：

2018年05月3日 16:07

后面的经验总结很赞！节省调试时间，避免了踩坑！

回复


15.

zhenhappy说道：

2018年05月3日 14:25

感谢旭哥提供思路, 我简单封装了个播放器, 献丑了
<https://github.com/zhenhappy/images-player>

回复


16.

广州网站建设说道：

2018年05月3日 11:46

很多时候都会遇到这些问题，学习了

回复


17.

Tong说道：

2018年05月3日 11:33

这和上面的第3、4种方法其实差别不大，性能上更好可能只是你的感觉

回复


18.

magicbing说道：

2018年05月3日 11:32

看云风博客说做梦幻系列时候, 鸣谢名单就是按帧播放图片做的视频效果

回复



19.

小媒体说道：

2018年05月2日 18:43

思路不错，如下功能如果有就更好了：

1. 边下载边播放

2. 快进快退

3. 声音，关键和快进快退要同步

不过这三个问题，业务开发者可以自己来实现。

回复

riophae说道：

2018年05月3日 11:41

1. 边下载边播放也可以做，主要是担心播到一半突然卡住，体验会很不好，倒不如预先都加载完再播

2. 和 3. 都能做

回复

小媒体说道：

2018年05月3日 19:19

小视频 网速好 完全可以全部加载完再播放，反之 如果没有边下边播，体验会更差。

3 具体怎么做，说下思路。

回复

20.

lmfyeee说道：

2018年05月2日 15:46

学习了学习了，前俩天刚好了解到这种原理。今天就刚好看到鑫哥已经写出来了

回复

21.

lucas_chuan说道：

2018年05月2日 11:01

突如其来的吻让我猝不及防惹

回复

22.

trance说道：

2018年05月2日 10:18

视频看笑了 代码学习了 效果达到了

回复

23.

twoer说道：

2018年05月1日 21:27

为啥不考虑直接用 canvas ？

回复

张 鑫旭说道：

2018年05月1日 21:32

DOM扩展更方便。

回复

win说道：

2018年05月8日 13:59

canvas和dom按功能区分开呢？

canvas只做视频展示，设置不可点击；其余的还用dom来处理

回复

24. 老白狼说道:
2018年05月1日 19:40
绝对学习了了, 谢谢
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](#) ⁽¹⁷⁸⁾
- » [粉丝群第1期CSS小测点评与答疑](#) ⁽¹¹²⁾
- » [未来必热: SVG Sprite技术介绍](#) ⁽¹¹¹⁾
- » [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](#) ⁽⁸³⁾
- » [Selectivizr-让IE6~8支持CSS3伪类和属性选择器](#) ⁽⁸⁰⁾
- » [CSS3下的147个颜色名称及对应颜色值](#) ⁽⁷⁸⁾
- » [小tips: 纯CSS实现打字动画效果](#) ⁽⁷²⁾
- » [写给自己看的display: flex布局教程](#) ⁽⁶⁹⁾
- » [分享三个纯CSS实现26个英文字母的案例](#) ⁽⁶⁹⁾

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

- [从天猫某活动视频不必要的3次请求说起](#)
- [canvas图形绘制之星空、噪点与烟雾效果](#)
- [基于clip-path的任意元素的碎片拼接动效](#)
- [canvas实现iPhoneX炫彩壁纸屏保外加pixi.js流体动效](#)
- [CSS CSS3实现3D开门动画效果](#)
- [回流与重绘: CSS性能让JavaScript变慢?](#)
- [翻译 - CSS Sprites:实用技术还是生厌之物?](#)
- [移动页面加速google的AMP和百度的MIP简介](#)
- [让所有浏览器支持HTML5 video视频标签](#)
- [揭秘视频网站video视频倍速播放的实现](#)
- [HTML5新增的form属性简介](#)

