

# HTML5 file API加canvas实现图片前端JS压缩并上传

这篇文章发布于 2017年07月30日, 星期日, 23:47, 归类于 [Canvas相关](#), [JS实例](#)。阅读 70351 次, 今日 56 次 [50 条评论](#)

by zhangxinxu from <http://www.zhangxinxu.com/wordpress/?p=6308>

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、图片上传前端压缩的现实意义

对于大尺寸图片的上传, 在前端进行压缩除了省流量外, 最大的意义是极大的提高了用户体验。

这种体验包括两方面:

1. 由于上传图片尺寸比较小, 因此上传速度会比较快, 交互会更加流畅, 同时大大降低了网络异常导致上传失败风险。
2. 最最重要的体验改进点: 省略了图片的再加工成本。很多网站的图片上传功能都会对图片的大小进行限制, 尤其是头像上传, 限制5M或者2M以内是非常常见的。然后现在的数码设备拍摄功能都非常出众, 一张原始图片超过2M几乎是标配, 此时如果用户想把手机或相机中的某个得意图片上传作为自己的头像, 就会遇到因为图片大小限制而不能上传的窘境, 不得不对图片进行再处理, 而这种体验其实非常不好的。如果可以在前端进行压缩, 则理论上对图片尺寸的限制是没有必要的。

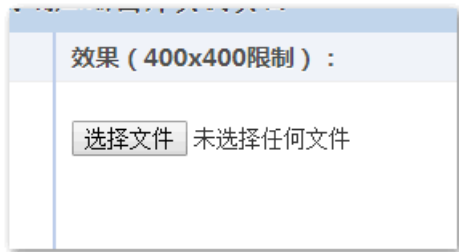
## 二、图片前端JS压缩并上传功能体验

特意制作了一个图片前端压缩并上传的完整demo, 您可以狠狠的点击这里: [使用canvas在前端压缩图片并上传demo](#)

进入demo会看到一个相貌平平的文件输入框:



啊, 不对, 应该是这张图:



点击文件选择框，我们不妨选一张尺寸比较大的图片，例如下面这种2M多的钓鱼收获照：



于是图片欸欸地传上去了：



此时我们点击最终上传完毕的图片地址，会发现原来2M多3000多像素宽的图片被限制为400像素宽了：



保存到本地会发现图片尺寸已经变成只有70K了：



以上就是图片前端压缩并上传demo的完整演示。

### 三、HTML5 file API加canvas实现图片前端JS压缩

要想使用JS实现图片的压缩效果，原理其实很简单，核心API就是使用 `canvas` 的 `drawImage()` 方法。

`canvas` 的 `drawImage()` 方法API如下：

```
context.drawImage(img, dx, dy);
context.drawImage(img, dx, dy, dwidth, dheight);
context.drawImage(img, sx, sy, swidth, sheight, dx, dy, dwidth, dheight);
```

后面最复杂的语法虽然看上去有9大参数，但不用慌，实际上可以看出就3个参数：

#### `img`

就是图片对象，可以是页面上获取的DOM对象，也可以是虚拟DOM中的图片对象。

#### `dx, dy, dWidth, dHeight`

表示在 `canvas` 画布上规划处一片区域用来放置图片，`dx, dy` 为canvas元素的左上角坐标，`dWidth, dHeight` 指canvas元素上用在显示图片的区域大小。如果没有指定 `sx, sy, swidth, sheight` 这4个参数，则图片会被拉伸或缩放在这片区域内。

#### `sx, sy, swidth, sheight`

这4个坐标是针对图片元素的，表示图片在 `canvas` 画布上显示的大小和位置。`sx, sy` 表示图片上 `sx, sy` 这个坐标作为左上角，然后往右下角的 `swidth, sheight` 尺寸范围图片作为最终在 `canvas` 上显示的图片内容。

`drawImage()` 方法有一个非常怪异的地方，大家一定要注意，那就是5参数和9参数里面参数位置是不一样的，这个和一般的API有所不同。一般API可选参数是放在后面。但是，这里的 `drawImage()` 9个参数时候，可选参数 `sx, sy, swidth, sheight` 是在前面的。如果不注意这一点，有些表现会让你无法理解。

下图为MDN上原理示意：

对于本文的图片压缩，需要用的是5个参数语法。举个例子，一张图片（假设图片对象是 `img`）的原始尺寸是4000\*3000，现在需要把尺寸限制为400\*300大小，很简单，原理如下代码示意：

```
var canvas = document.createElement('canvas');
var context = canvas.getContext('2d');
canvas.width = 400;
canvas.height = 300;
// 核心JS就这个
context.drawImage(img,0,0,400,300);
```

把一张大的图片，直接画在一张小小的画布上。此时大图片就天然变成了小图片，压缩就这么实现了，是不是简单的有点超乎想象。

当然，若要落地于实际开发，我们还需要做些其他的工作，就是要解决图片来源和图片去向的问题。

## 1. 如何把系统中图片呈现在浏览器中？

HTML5 file API可以让图片在上传之前直接在浏览器中显示，通常使用 `FileReader` 方法，代码示意如下：

```
var reader = new FileReader(), img = new Image();
// 读文件成功的回调
reader.onload = function(e) {
    // e.target.result就是图片的base64地址信息
    img.src = e.target.result;
};
eleFile.addEventListener('change', function (event) {
    reader.readAsDataURL(event.target.files[0]);
});
```

于是，包含图片信息的 `context.drawImage()` 方法中的 `img` 图片就有了。

## 2. 如果把canvas画布转换成img图像

`canvas` 天然提供了2个转图片的方法，一个是：

### `canvas.toDataURL()`方法

语法如下：

```
canvas.toDataURL(mimeType, qualityArgument)
```

可以把图片转换成base64格式信息，纯字符的图片表示法。

其中：

`mimeType` 表示 `canvas` 导出出来的 `base64` 图片的类型，默认是png格式，也即是默认值是 `'image/png'`，我们也可以指定为jpg格式 `'image/jpeg'` 或者webp等格式。`file` 对象中的 `file.type` 就是文件的mimeType类型，在转换时候正好可以直接拿来用（如果有file对象）。

`qualityArgument` 表示导出的图片质量，只要导出为 `jpg` 和 `webp` 格式的时候此参数才有效果，默认值是 `0.92`，是一个比较合理的图片质量输出参数，通常情况下，我们无需再设定。

### `canvas.toBlob()`方法

语法如下：

```
canvas.toBlob(callback, mimeType, qualityArgument)
```

可以把canvas转换成Blob文件，通常用在文件上传中，因为是二进制的，对后端更加友好。

和 `toDataURL()` 方法相比，`toBlob()` 方法是异步的，因此多了个 `callback` 参数，这个 `callback` 回调方法默认的第二个参数就是转换好的 `blob` 文件信息，本文demo的文件上传就是将 `canvas` 图片转换成二进制的 `blob` 文件，然后再 `ajax` 上传的，代码如下：

```
// canvas转为blob并上传
canvas.toBlob(function (blob) {
  // 图片ajax上传
  var xhr = new XMLHttpRequest();
  // 开始上传
  xhr.open("POST", 'upload.php', true);
  xhr.send(blob);
});
```

于是，经过“图片→canvas压缩→图片”三步曲，我们完成了图片前端压缩并上传的功能。

更加完整的核心代码请参见[demo页面](#)的左侧，如果对其他交互代码也感兴趣，请参考页面源代码。

## 四、结束语

就在几个月前刚写过一篇文章“[使用canvas在前端实现图片水印合成](#)”，实际上所使用的技术和套路和本文是如出一辙的，也是“图片→canvas水印→图片”三步曲，区别在于水印合成是连续执行两次 `context.drawImage()` 方法，一次是原图一次水印图片，以及最后转换成图片的时候什么是 `toDataURL()` 方法，其他代码逻辑和原理都是一样的。

由此及彼，利用同样的原理和代码逻辑，我们还可以实现其它很多以前前端不太好实现的功能，比方说图片的真剪裁效果，所谓“真剪裁”指不是使用个 `overflow:hidden` 或者 `clip` 这些CSS属性的“伪剪裁”，而是真正意义上就这么大区域图片信息。甚至配合一些前端算法，我们可以直接在前端进行人脸识别，图片自动美化等一系列功能再上传等等。

原理都是一样的，都是利用 `canvas` 作为中间媒介进行处理。

好，以上就是本文的全部内容，感谢阅读，欢迎纠错，欢迎交流！

《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](#)。



- XMLHttpRequest实现HTTP协议下文件上传断点续传
- HTML input type=file文件选择表单元素二三事
- 纯前端实现可传图可字幕台词定制的GIF表情生成器
- 小tips: 纯前端JS读取与解析本地文本类文件
- 直接剪切板粘贴上传图片的前端JS实现
- Ajax Upload多文件上传插件翻译及中文演示
- 理解DOMString、Document、FormData、Blob、File、ArrayBuffer数据类型
- 基于HTML5的可预览多图片Ajax上传
- 小tips:使用canvas在前端实现图片水印合成
- 原来浏览器原生支持JS Base64编码解码
- 新版无图片版zxxbox jQuery弹出框插件

分享到: 1

标签: Ajax, Blob, canvas, File, FileReader, HTML5, 上传, 图片压缩

## 发表评论（目前50条评论）

名称 (必须)

邮件地址(不会被公开) (必须)

网站

提交评论

- 我自向天笑说道:  
2018年11月5日 15:20

请教您一个问题：将苹果手机竖屏拍的照片用canvas绘制出来的时候为何会有一部分图片被剪切掉绘制不出来？我已研究多天，尝试了很多方法，修改参数或者旋转图片均不奏效

回复



- 张 鑫旭说道:  
2018年11月6日 00:27

安卓机器没有这个问题吗？

回复



- 我自向天笑说道:  
2018年11月6日 09:16

就测试的小米，华为，vivo这三款安卓机而已，未发现问题，苹果用原生js实现也不存在这个问题。我用的是ionic3框架，在这个框架下就会出问题，经过多次测试，怀疑是naturalWidth和naturalHeight与图片的实际不符，同样的图片，两种写法得到的结果恰恰相反，现在还在想办法处理。

回复



noahlam说道:

2019年01月12日 22:34

苹果照片好像分辨率都是1，而普通的照片是72，会不会是这一个原因导致 naturalWidth 和 naturalHeight 计算不准确？



2. abaddonpoet说道:

2018年08月17日 10:34

人脸识别已经有js库了，opencv和ccv。我用过了。。感觉超棒。。就是挨千刀的索尼手机照片exif信息总是不随主流。。方向老是反的， -

[回复](#)



3. 前端Logic说道:

2018年07月18日 10:55

用toBlob得到流数据后，上传时显示fileName为”blob”。

前端能自己设置这个上传的文件名吗

[回复](#)



前端Logic说道:

2018年07月18日 11:09

我已经知道了，鑫哥。

formData.append(“image”, blob, “xxx.jpg”); 第3个参数就能设置fileName了

[回复](#)



4. 前端中白说道:

2018年06月7日 22:00

学到了！！感激不尽

[回复](#)



5. C# 后台怎么接收啊说道:

2018年05月24日 16:22

C# 后台怎么接收啊

[回复](#)



6. 大神说道:

2018年05月22日 11:41

大神大神，快收下我的膝盖！！！！

[回复](#)



7. BaoMax说道:

2018年05月9日 11:40

canvas.toBlob的方法会阻塞页面，但是它不是异步的吗，大佬知道因为啥吗

[回复](#)



张 鑫旭说道:

2018年05月9日 13:56

因为一瞬间性能开销比较大~

[回复](#)



**BaoMax**说道:

2018年05月9日 16:33

我压缩一个十几兆的图片耗时5-10s,这期间页面都是卡死状态,是不是这么大的不能用这种压缩了,前端还有别的压缩方法吗?

[回复](#)

**张 鑫旭**说道:

2018年05月9日 17:05

你把这种图片小尺寸绘制在canvas上,再转blob还卡?



8. **vcxiaohan**说道:

2018年05月8日 14:39

这个应该是缩放尺寸吧,压缩应该是对图片质量的压缩  
canvas.toDataURL("image/jpeg",[压缩比率])

[回复](#)



9. **图片压缩问题**说道:

2018年04月15日 00:25

被压缩的图片上传后模糊失真

[回复](#)



10. **Valk**说道:

2018年04月12日 16:54

目前正在实现一个功能,利用canvas合成图片并上传到服务器。然后在微信端展示出来的。现在的问题是在canvas上传到服务器这一步出了问题。谷歌浏览器是没问题的,然而搜狗浏览器不支持toBlob(想要各浏览器兼容),因此我使用了toDataURL,得到base64的数据,然后 blob = new Blob([ 图片的base64数据 ],'image/png');然后将这个blob对象传到服务器。一切都正常,然而后台得到的图片展示不出来了。猜测是这样转成blob,数据有丢失? 请问大神有没有其他方法可以将canvas画出来的图片传到服务器(不支持toBlob的情况下)?

[回复](#)



11. **Faith**说道:

2018年03月19日 17:26

已按您的说的成功写出案例,但是在使用中,发现图片模糊的比较模糊,就算把qualityArgument设为1,得到的图片也很模糊,比如拍一张A4,等压缩好上传之后,字基本上都是看不清楚的。请问,还有没有什么比较好的解决方案。

[回复](#)



**Flcwl**说道:

2018年04月2日 08:44

把drawImage()的 dWidth, dHeight两个参数调大试试,应该可以让像素高点,当然空间也就占的更大了。

[回复](#)



12. **IMIN**说道:

2018年03月17日 18:02

鑫大王,我点你那个上传,不小心把我驾驶证上传了。。。你可别拿我驾驶证扣分啊。。QAQ

[回复](#)



**toya**说道:

2018年04月8日 10:53

这位兄弟,你怕不是要笑死我。



[回复](#)

云淡、风轻说道:

2018年04月9日 11:36

兄弟操作可以呀，给你一波666

[回复](#)



13. **IE10都兼容不了**说道:

2018年02月16日 00:07

IE中不兼容

file.type.indexOf("image") 始终为 -1

canvas.toBlob(function (blob) { ... ... 在IE中出错

换了几个JQ（包括最新版的）都不兼容

[回复](#)



14. 白铭说道:

2018年01月27日 17:34

upload.php 代码 能给一下吗?

[回复](#)



15. 楚说道:

2018年01月11日 17:17

支持多个图片上传吗

[回复](#)



16. 压缩后的图片很模糊说道:

2017年12月28日 15:02

压缩后的图片很模糊，应该是没处理高倍屏

[回复](#)



**Faith**说道:

2018年03月19日 17:44

那你怎么实现压缩之后，图片还是清晰的呀？我现在也是，压缩完之后，图片的内容一点也看不清楚。

[回复](#)



17. **winlion**说道:

2017年11月26日 22:39

参考楼主代码封装了一个demo

<http://www.imwinlion.com/archives/1>

前端js 后端go

[回复](#)



18. **xiaoxin**说道:

2017年11月3日 16:30

最近使用FileReader出现一个问题，在小米2s手机上拍照上传中，读取成功后获取不到编码result值，弹出是data:，后面什么都没有，用别的手机测试没问题，而且是只有拍照是这样，选择相册是可以获取到该值的，现在也不知道是什么原因造成的，不知道大神遇到过没有





[回复](#)

19. **vc2001895**说道：  
2017年10月27日 14:19

求处理页的upload.php demo

[回复](#)



20. **请问一下**说道：  
2017年09月30日 17:04

为什么canvas建立之后要  
// 清除画布  
context.clearRect(0, 0, targetWidth, targetHeight);  
这里清除了一下。  
但建立之后，应该是个白屏（或透明色）  
即便有颜色，直接覆盖绘制，结果不是也一样的吗？  
求解答。

[回复](#)



21. **小白**说道：  
2017年09月29日 12:51

H5在真机上好像不兼容

[回复](#)



22. **boyang**说道：  
2017年09月25日 17:35

裁剪尺寸的图转blob对象 size 会比 原图转blob对象的 size 大很多是为什么？

[回复](#)



23. **Faith**说道：  
2017年09月12日 13:07

移动端用canvas的时候，为了保证适配，用的rem为单位布局，可是canvas的width不支持rem单位，所以移动端如何做？

[回复](#)



**zzxqd**说道：  
2017年09月25日 17:04

页面初始化的时候用js动态改啊

[回复](#)



24. **xiaobin**说道：  
2017年08月28日 15:50

很受益

[回复](#)



25. **yuye**说道：  
2017年08月28日 11:34

能 公布后台 upload.php 代码吗，我想看看后台怎么接收的

[回复](#)



26. **kml1200**说道:  
2017年08月26日 15:09
- 请问:  
使用FileReader将文件转换为base64  
和  
使用window的URL工具的createObjectURL方法将文件转换成blob对象,  
让图片在上传之前在浏览器中显示, 有什么差别呢, 哪种方法较好?
- [回复](#)
27. **益阳**说道:  
2017年08月23日 23:00
- 把一张大的图片, 直接画在一张小小的画布上。用户体验不好呀。
- [回复](#)
28. **leega0**说道:  
2017年08月15日 18:09
- 在Mac的safari下, TypeError: canvas.toBlob is not a function. (In 'canvas.toBlob', 'canvas.toBlob' is undefined)  
看来不支持canvas这个方法
- [回复](#)
- jackma**说道:  
2017年08月23日 11:17
- 有polyfill的。  
<https://github.com/blueimp/JavaScript-Canvas-to-Blob>
- [回复](#)
29. **JUNE**说道:  
2017年08月15日 17:30
- 你好! 请问upload.php应该怎么接收传过来的数据, upload.php文件传我一份呗; 谢啦
- [回复](#)
30. **摩缺**说道:  
2017年08月10日 10:19
- 我做微信公众h5注册页面需要图片上传, 也出现上传图片过大问题, 后面采用了canvas压缩的办法, 但是出现一个新的问题, 压缩前的图片过大, 压缩时会出现浏览器崩溃闪退问题, 目前测试出较大图片压缩, 以及低配的手机(小米4、Nexus 5等)出现此类问题频率较高
- [回复](#)
- lilei**说道:  
2017年11月14日 10:13
- 我也遇到像你类似的问题, 你解决了吗, 如果解决的
- [回复](#)
31. **wusy**说道:  
2017年08月9日 23:43
- 后台代码 upload.php  
我用java怎么写  
以前都是 后台接受file 然后上传源文件 现在后台怎么写呢?  
以前

—32525982912960643881027537595  
Content-Disposition: form-data; name="upload"; filename="truck.png"  
Content-Type: image/png  
—32525982912960643881027537595—

现在：没有上面的内容 后台咋接受

[回复](#)

32. **andy**说道：

2017年08月7日 11:41

为什么没有demo下载呢？因为里面使用到了upload.php，那就要配合后端使用。能提供这个php的demo么？

[回复](#)



33. **系心**说道：

2017年07月31日 10:51

“使用canvas在前端实现图片水印合成”那个超链接地址放错了……

[回复](#)



张鑫旭说道：

2017年07月31日 21:43

好的，感谢反馈~

[回复](#)



34. **wyl**说道：

2017年07月31日 10:45

不用注册也可以填吗

[回复](#)



35. **ptrsvltns**说道：

2017年07月31日 04:40

沙发!!!

[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](#) <sup>(190)</sup>
- » [未来必热：SVG Sprite技术介绍](#) <sup>(119)</sup>
- » [粉丝群第1期CSS小测点评与答疑](#) <sup>(115)</sup>
- » [HTML5终极备忘大全（图片版+文字版）](#) <sup>(93)</sup>
- » [让所有浏览器支持HTML5 video视频标签](#) <sup>(86)</sup>

- » [Selectivizr-让IE6~8支持CSS3伪类和属性选择器](#) <sup>(82)</sup>
- » [CSS3下的147个颜色名称及对应颜色值](#) <sup>(79)</sup>
- » [视区相关单位vw, vh..简介以及可实际应用场景](#) <sup>(76)</sup>
- » [写给自己看的display: flex布局教程](#) <sup>(76)</sup>
- » [小tips: 纯CSS实现打字动画效果](#) <sup>(76)</sup>

#### 今年热议

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

#### 猜你喜欢

- [XMLHttpRequest实现HTTP协议下文件上传断点续传](#)
- [HTML input type=file文件选择表元素二三事](#)
- [纯前端实现可传图可字幕台词定制GIF表情生成器](#)
- [小tips: 纯前端JS读取与解析本地文本类文件](#)
- [直接剪切板粘贴上传图片的前端JS实现](#)
- [Ajax Upload多文件上传插件翻译及中文演示](#)
- [理解DOMString、Document、FormData、Blob、File、ArrayBuffer数据类型](#)
- [基于HTML5的可预览多图片Ajax上传](#)
- [小tips:使用canvas在前端实现图片水印合成](#)
- [原来浏览器原生支持JS Base64编码解码](#)
- [新版无图片版zxxbox jQuery弹出框插件](#)