[移动端图片上传解决方案localResizeIMG先压缩后ajax无刷新上传](http://www.cnblogs.com/manongxiaobing/p/4720568.html)

网址：http://www.cnblogs.com/manongxiaobing/p/4720568.html

现在科技太发达，移动设备像素越来越高，随便一张照片2M+，但是要做移动端[图片上传](http://www.devdo.net/tag/%e5%9b%be%e7%89%87%e4%b8%8a%e4%bc%a0" \o "View all posts in 图片上传" \t "_blank)和pc上略有不同，移动端你不能去限制图片大小，让用户先处理图片再上传，这样不现实。所以理解的解决方案就是在上传先进行图片[压缩](http://www.devdo.net/tag/%e5%8e%8b%e7%bc%a9" \o "View all posts in 压缩" \t "_blank)，然后再把压缩后的图片上传到服务器。

一翻google之后，发现了[localResizeIMG](http://www.devdo.net/tag/localresizeimg" \o "View all posts in localResizeIMG" \t "_blank)，它会对图片进行压缩成你指定宽度及质量度并转换成base64图片格式，那么我们就可以把这个base64通过ajax传到后台，再进行保存，先压缩后上传的目的就达到了。

**处理过程**

1. LocalResizeIMG压缩图片
2. AjaxPost图片base64到后台
3. 后台接收base64并保存，返回状态

**前台代码**

重点，引用LocalResizeIMG.js（插件主体）及mobileBUGFix.mini.js（移动端的补丁）

<!DOCTYPE HTML>

<html>

<head>

<meta charset="UTF-8">

<title>移动端图片上传解决方案localResizeIMG先压缩后ajax无刷新上传</title>

<meta name="description" content="" />

<meta name="viewport" content="width=device-width , initial-scale=1.0 , user-scalable=0 , minimum-scale=1.0 , maximum-scale=1.0" />

<script type='text/javascript' src='js/jquery-2.0.3.min.js'></script>

<script type='text/javascript' src='js/LocalResizeIMG.js'></script>

<script type='text/javascript' src='js/patch/mobileBUGFix.mini.js'></script>

<style type="text/css">

body{font-family:"微软雅黑"}

.uploadbtn{ display:block;height:40px; line-height:40px; color:#333; text-align:center; width:100%; background:#f2f2f2; text-decoration:none; }

.imglist{min-height:200px;margin:10px;}

.imglist img{width:100%;}

</style>

</head>

<body>

<div style="width:500px;margin:10px auto; border:solid 1px #ddd; overflow:hidden; ">

<input type="file" id="uploadphoto" name="uploadfile" value="请点击上传图片" style="display:none;" />

<div class="imglist"></div>

<a href="javascript:void(0);" onclick="uploadphoto.click()" class="uploadbtn">点击上传文件</a>

</div>

<div style="text-align:center;margin-top:50px;">@ <a href="http://www.devdo.net/">码农小兵,专注web开发 欢迎投稿</a></div>

</body>

</html>

**Js部份，localResizeIMG及Ajax提交部份**

**使用方法**

$('input:file').localResizeIMG({

width: 400,//宽度

quality: 1,//质量

success: function (result) {

result.base64/result.clearBase64

}

});

**localResizeIMG参数：**

* width:缩略图宽度
* quality:图片质量，0—1，越大越好

**localResizeIMG返回值**

* result.base64:带图片类型的base64编码，可直接用于img标签的src，如“…2wBDAAYEBQYFBAY”;
* result.clearBase64:不带图片类型的编码，如“/9j/4AAQSkZJRgABAQAAAQABAAD/…2wBDAAYEBQYFBAY”

$(document).ready(function(e) {

$('#uploadphoto').localResizeIMG({

width: 400,

quality: 1,

success: function (result) {

var submitData={

base64\_string:result.clearBase64,

};

$.ajax({

type: "POST",

url: "upload.php",

data: submitData,

dataType:"json",

success: function(data){

if (0 == data.status) {

alert(data.content);

return false;

}else{

alert(data.content);

var attstr= '<img src="'+data.url+'" alt="" />';

$(".imglist").append(attstr);

}

},

complete :function(XMLHttpRequest, textStatus){

},

error:function(XMLHttpRequest, textStatus, errorThrown){ //上传失败

alert(XMLHttpRequest.status);

alert(XMLHttpRequest.readyState);

alert(textStatus);

}

});

}

});

});

**保存文件**

在上面一步中，我们把result.clearBase64通过Ajax传入到upload.php中，接下来我们就要在upload.php中接收base64参数，把它转换成img文件保存来服务器中，并给出提示。

$base64\_string = $\_POST['base64\_string'];

$savename = uniqid().'.jpeg';//localResizeIMG压缩后的图片都是jpeg格式

$savepath = 'images/'.$savename;

$image = base64\_to\_img( $base64\_string, $savepath );

if($image){

echo '{"status":1,"content":"上传成功","url":"'.$image.'"}';

}else{

echo '{"status":0,"content":"上传失败"}';

}

function base64\_to\_img( $base64\_string, $output\_file ) {

$ifp = fopen( $output\_file, "wb" );

fwrite( $ifp, base64\_decode( $base64\_string) );

fclose( $ifp );

return( $output\_file );

}

**不足之处**

* localResizeIMG压缩后的图片模式都是jpeg，不能保证原有格式。
* 当图片宽度小于localResizeIMG设置的width参数时，图片会被拉申，从而引起图片失真（比如width高为600，图片只在400px时，压缩后的图片就变成了600px,图片尺寸变大了，会失真），不知道大家有没有什么好的解决方法。