

SOFTWARE

RESIZE AND PRESERVE IMAGE ORIENTATION BEFORE UPLOAD

24.07.2016

I need to downscale images on mobile phones to save bandwidth. Here is my code based on <http://www.codeforest.net/html5-image-upload-resize-and-crop>. No need for any complicated plugins. Take care of some security holes...

```
<html>
<head>
<script src="jquery-1.11.1.min.js"></script>
<?php if ($_POST) {
    $img = $_POST['image'];
    $img = str_replace('data:image/jpeg;base64,', '', $img);
    $img = str_replace(' ', '+', $img);
    $data = base64_decode($img);
    $file = 'cache/' . uniqid() . '.jpg';
    file_put_contents($file, $data);
}
?>
</head>

<body>
<form>
<input type="text" value="" name="text" id="text"/>
<input type="file" name="filesToUpload[]" id="filesToUpload"
multiple="multiple" />
<input type="button" id="button" value="submit">
</form>

<script>
$('#button').click(function(){

    if (window.File && window.FileReader && window.FileList &&
```

```
window.Blob) {
    var files = document.getElementById('filesToUpload').files;
    for(var i = 0; i < files.length; i++) {
        resizeAndUpload(files[i]);
    }
}
else {
    alert('not supported');
}
});
```

```
function resizeAndUpload(file) {
    var reader = new FileReader();
    reader.onloadend = function() {
        var tempImg = new Image();
        tempImg.src = reader.result;
        tempImg.onload = function() {
            var max_width = 500;
            var max_height = 500;
            var tempW = tempImg.width;
            var tempH = tempImg.height;
            if (tempW > tempH) {
                if (tempW > max_width) {
                    tempH *= max_width / tempW;
                    tempW = max_width;
                }
            }
            else {
                if (tempH > max_height) {
                    tempW *= max_height / tempH;
                    tempH = max_height;
                }
            }
        }
        var canvas = document.createElement('canvas');
        canvas.width = tempW;
        canvas.height = tempH;
        var ctx = canvas.getContext("2d");
        ctx.drawImage(this, 0, 0, tempW, tempH);
        var dataURL = canvas.toDataURL("image/jpeg");
        var xhr = new XMLHttpRequest();
        xhr.onreadystatechange = function() {
```

```
    if (xhr.readyState == 4 && xhr.status == 200) {
        location.reload();
    }
};
xhr.open('POST', 'index.php', true);
xhr.setRequestHeader("Content-type","application/x-www-form-urlencoded");
var input = document.getElementById("text");
var inputData = encodeURIComponent(input.value);
var data = 'text=' + inputData + '&image=' + dataURL;
xhr.send(data);
}
}
</script>

</body>
</html>
```

The big issue when running that with multiple phones is image orientation. To save computing power many mobiles are just save their sensor data without caring about too much about orientation as long as orientation is written in the exif tag. Here are the 8 possible orientations

ORIENTATION		
	TOP	LEFT
1	top	left
2	top	right
3	bottom	right
4	bottom	left
5	left	top
6	right	top
7	right	bottom
8	left	bottom

I therefore added canvas rotation / flipping after getting image orientation from <https://github.com/blueimp/JavaScript-Load-Image>. The revised function now looks like



```
function resizeAndUpload(file) {
  var reader = new FileReader();
  reader.onloadend = function() {
    var can = document.createElement('canvas');
    var ctx = can.getContext("2d");
    var thisImage = new Image();
    thisImage.src = reader.result;
    thisImage.onload = function() {
      var max_width = 1500;
      var max_height = 1500;
      var tempW = thisImage.width;
      var tempH = thisImage.height;
      if (tempW > tempH) {
        if (tempW > max_width) {
          tempH *= max_width / tempW;
          tempW = max_width;
        }
      }
      else {
        if (tempH > max_height) {
          tempW *= max_height / tempH;
          tempH = max_height;
        }
      }
      can.width = tempW;
      can.height = tempH;
      ctx.save();
      var width = can.width; var styleWidth = can.style.width;
      var height = can.height; var styleHeight = can.style.height;
      if (orientation>1) {
        if (orientation > 4) {
          can.width = height; can.style.width = styleHeight;
          can.height = width; can.style.height = styleWidth;
        }
        switch (orientation) {
          case 2: ctx.translate(width, 0);      ctx.scale(-1,1); break;
          case 3: ctx.translate(width,height);  ctx.rotate(Math.PI);
break;
          case 4: ctx.translate(0,height);      ctx.scale(1,-1); break;
          case 5: ctx.rotate(0.5 * Math.PI);   ctx.scale(1,-1); break;
          case 6: ctx.rotate(0.5 * Math.PI);
```

```
ctx.translate(0,-height); break;
    case 7: ctx.rotate(0.5 * Math.PI);
ctx.translate(width,-height); ctx.scale(-1,1); break;
    case 8: ctx.rotate(-0.5 * Math.PI);
ctx.translate(-width,0); break;
    }
}
ctx.drawImage(thisImage,0,0,tempW,tempH);
ctx.restore();
}
var dataURL = canvas.toDataURL("image/jpeg");
var xhr = new XMLHttpRequest();
xhr.onreadystatechange = function() {
    if (xhr.readyState == 4 && xhr.status == 200) {
        location.reload();
    }
};
xhr.open('POST', 'index.php', true);
xhr.setRequestHeader("Content-type","application/x-www-form-urlencoded");
var input = document.getElementById("text");
var inputData = encodeURIComponent(input.value);
var data = 'text=' + inputData + '&image=' + dataURL;
xhr.send(data);
}
}
```