



链滴

java压缩图片生成缩略图，并添加水印

作者: [crick77](#)

原文链接: <https://ld246.com/article/1450234882493>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

起因

在编写的cms系统中，提供了图片上传功能，但是用户经常(或者说从不)按照推荐的比例上传图片，然这不是主要问题，关键点在于，经常会上传一些超大分辨率的图片，严重影响整个网站的加载速度。

面对这个问题，除了限制上传图片的大小外，是否还有更自动化的解决方案？

待解决问题：

1. 图片长宽比
2. 图片大小
3. 图片格式
4. 水印(用户要求)

关于添加水印的问题，其实用户希望制作一款不能被下载图片的网站，但个人能力有限，实在不能胜。于是用户提出水印要求，本身可能会觉得水印是本地图片操作的职责，但为了体验系统的自动化，定增加自动添加水印功能。

解决方案

java 的io操作一直是我较为困扰的部分。原因无它，因为可以实现的方式实在是太多了，在数十个工类和数个知名类库中该如何选择？而java在图像处理方面，私以为并不占有优势。

经过多方比较，适用，最终采用了[thumbnailator](#)这一开源解决方案

thumbnailator

thumbnailator是一个开源的java生成缩略图类库，没有依赖其他外部库，不需要自己去学习Image I O API, Java 2D API, image processing, image scaling techniques, BufferedImages, Graphics D 等技术，可以简单方便的实现这些功能。

来源

thumbnailator是一个简单的jar文件，也支持通过maven进行依赖管理

```
<dependency>
  <groupId>net.coobird</groupId>
  <artifactId>thumbnailator</artifactId>
  <version>0.4.8</version>
</dependency>
```

使用方法

thumbnailator提供的功能简单、强大，且API规则良好，可以容易的理解接口含义。下面针对上面问题，介绍一下thumbnailator的主要功能。

重置图片大小

```
Thumbnails.of(new File("original.jpg"))
    .size(640, 480)
    .outputFormat("jpg")
    .toFiles(Rename.NO_CHANGE);
```

`new File("original.jpg")` 可以是`listFiles()`来实现对目录下所有文件的操作，也可以直接输入String类"original.jpg"

`toFiles`表示文件输出位置，单个文件可以自定义指定文件路径。

在处理多文件时，可以根据Rename提供的方法Rename.NO_CHANGE表示变更当前图片文件，还有`ename.PREFIX_DOT_THUMBNAIL`等增加前缀、后缀等方式，此时会保留原文件和生成后的图片文。

`outputFormat`现支持`ImageIO.getWriterFormatNames()`提供的格式类型，本测试环境为jdk1.6.0_8，提供的类型有BMP bmp jpg JPG wbmp jpeg png PNG JPEG WBMP GIF gif

生成一个旋转90°带水印的缩略图

```
Thumbnails.of(new File("original.jpg"))
    .size(160, 160)
    .rotate(70)
    .watermark(Positions.BOTTOM_RIGHT, ImageIO.read(new File("watermark.png")), 0.5f)
    .outputQuality(0.8)
    .toFile(new File("image-with-watermark.jpg"));
```

`rotate`表示图片旋转角度，如果不是进行直角($n \times 90^\circ$)旋转的话，图片的size会发生变化。

`watermark`指定了水印，通过`Positions`指定水印位置，`BufferedImage`指定水印图片，第三个参数定了水印的不透明性，范围为(0|1.0f),1.0f为不透明。

`outputQuality` 指定输出图片的质量，范围同样为(0|1.0f)。

除了`toFile`，Thumbnails还可以将图片输出为`OutputStream(toOutputStream)`和`BufferedImage(sBufferedImage)`，或者是指定的输出目录(`toFiles`)。

按比例缩略图片

```
Thumbnails.of(new File("original.jpg"))
    .scale(0.2)
    .outputFormat("jpg")
    .toFiles(Rename.NO_CHANGE);
```

`scale`提供了按照比例缩略图片的方法，也可以通过大于1的数字扩大图片。

存在的问题

在测试环境中，发现水印图片过大导致显示不全和旋转图片后的图片比例变更等问题，都是因为使用不当。

在生产环境中，暂时未遇到使用问题。如果有问题，会在这里指出，并提供解决方案。

结语

thumbnailator是难得的让我通过java简单的使用io和图像处理的工具类，很喜欢它提供api方式，正学习源码。