

Linux 图像处理 (convert)

作者: [lbb4511](#)

原文链接: <https://ld246.com/article/1561841141342>

来源网站: [链滴](#)

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

convert 命令可以用来转换图像的格式，支持 JPG, BMP, PCX, GIF, PNG, TIFF, XPM 和 XWD 等类，下面举几个例子：

```
convert xxx.jpg xxx.png // 将jpeg转成png文件
convert xxx.gif xxx.bmp // 将gif转换成bmp图像
convert xxx.tiff xxx.pcx // 将tiff转换成pcx图像
convert -resize 1024x768 xxx.jpg xxx1.jpg // 将图像的像素改为1024*768，注意1024与768间是小写字母x
convert -sample 50% x50% xxx.jpg xxx1.jpg // 将图像的缩减为原来的50%*50%
convert -rotate 270 sky.jpg sky-final.jpg // 将图像顺时针旋转270度
convert -fill black -pointsize 60 -font helvetica -draw 'text 10,80 "Hello, World!" ' hello.jpg hello-world.jpg
```

1. 批量图像格式转换

- 如果想将某目录下的所有 jpg 文件转换为 png 文件，只要在命令行模式下输入：

```
for %f in (*.jpg) do convert "%f" "%~nf.png"
```

2. 对所有图像进行同一操作

- 譬如，批量生成某目录下所有 PNG 图像文件的缩略图(大小为 80×40)：

```
for %f in (*.png) do convert "%f" -sample 80x40 "%~nf_sample.png"
```

- 类似的，将某目录下所有 PNG 图像旋转 90 度的操作为：

```
for %f in (*.png) do convert "%f" -rotate 90 "%~nf_rotate.png"
```

- 还可以进行批量裁剪、淡化、抖动、炭化、加边框、圆角等等一系列操作，具体可参考：[通过命令行处理图形](#)

3. 在图像上加上文字说明

- 如果你有大量图片需要发布，在所有图片上加上版权说明是很明智的做法。用 ImgeMagick 可很容易的实现：

```
convert 1.png -fill white -pointsize 13 -draw "text 10,15 'lifesinger 2006 ' " 2.png
```

- 可以用-font 指定字体，这时需要安装 Ghostscript 支持：<http://www.cs.wisc.edu/~ghost/>
- 还可以用 composite 命令在所有图片上加上水印，有兴趣的看这里：~~<http://www.imagemagick.org/script/composite.php>~~

一些图像处理命令

- convert

转换图像格式和大小，模糊，裁剪，驱除污点，抖动，临近，图片上画图片，加入新图片，生成缩略图等。

- identify

描述一个或较多图像文件的格式和特性。

- mogrify

按规定尺寸制作一个图像，模糊，裁剪，抖动等。Mogrify 改写最初的图像文件然后写到一个不同的像文件。

- composite

根据一个图片或多个图片组合生成图片。

- montage

创建一些分开的要素图像。在含有要素图像任意的装饰图片，如边框、结构、图片名称等。

- compare

在算术上和视觉上评估不同的图片及其它的改造图片。

- display

如果你拥有一个 X server 的系统，它可以按次序的显示图片

- animate

利用 X server 显示动画图片

- import

在 X server 或任何可见的窗口上输出图片文件。你可以捕获单一窗口，整个的荧屏或任何荧屏的矩部分。

- conjure

解释执行 MSL (Magick Scripting Language) 写的脚本。

缩略图

- 生成一个 100×20 的缩略图

```
convert -sample 100x20 input.jpg output.jpg
```

- 更好的方法是用等比例缩放，像这样，统一生成 1 / 4 的缩略图

```
convert -sample 25%x25% input.jpg output.jpg
```

- 如果写成脚本，就是像这个样子

```
for img in `ls *.jpg`  
do  
    convert -sample 25%x25% ${img} thm${img}  
done
```

- 加注图片

```
convert -font fonts/font.ttf -stroke color -fill color -pointsize size -draw 'text 10,10 "String" in  
ut.jpg output.jpg
```

- -font 指定字体，因为这样我加注文字，
- -stroke 描边用的颜色，
- -fill 填充用的颜色，这里用 none 就可以画出空心字了，
- -pointsize 加注字体大小，像素数，
- -draw 是用来画，这里是文字，下面的位置 10，10 是以图片左上角为原点坐标的

• 向图像添加文本注释有时您需要向图像添加文本注释。例如，假设您的公司拥有标准的名片图像，希望在将名片发送到打印机之前将每个雇员的详细信息都添加到名片上面。另一个示例是为通过您网上的在线课程的用户生成表示证书 (presentation certificate) 。

- 您可以使用下列命令行，为该图注释一些标识信息：

```
convert -font helvetica -fill white -pointsize 36 -draw 'text 10,50 "Floriade 2002, Canberra, Aus
```

ralia" floriade.jpg comment.jpg

迄今为止，这是我在本文中所展示的最复杂的 convert 命令行了，因此我将花些时间来解释它。

- -font helvetica 将注释的字体设置为 Helvetica。也可以在此处指定字体文件的路径。这个示例图像添加了标记，这样未经许可其它网站就不能再使用该图像了，但它是使用位于非标准位置的字体完成该任务的：

```
convert -font fonts/1900805.ttf -fill white -pointsize 36 -draw 'text 10,475 "stillhq.com" floriade.jpg stillhq.jpg
```

- -fill white 用白色而不是标准的黑色来填充字母。

- -pointsize 36 以点为单位指定字母的大小。一英寸等于 72 点。

- -draw 'text 10,50 "...'" 是一组绘图命令，在本例中是移动到位置 10, 50，然后绘制出双引号中文本。使用单引号是因为如果需要绘制多个字，则绘图命令中需要使用双引号，而您不能在双引号中用双引号。

- 在一次 ImageMagick 调用中执行多条命令

您已经看到了将命令与注释示例联系起来的示例。但是，可以将本文中提到的任意 ImageMagick 命令链接起来。例如，也许我们希望制作某图像的缩略图，然后对它应用发散。在发散发生之后，我们将用炭笔效果：

```
convert -sample 25%x25% -spread 4 -charcoal 4 input.jpg output.jpg
```

- 用 convert 给图片加边框

```
convert -raise 5x5 input.jpg output.jpg
```

```
convert +raise 5x5 input.jpg output.jpg
```

以上命令分别用 - , + 边缘颜色的来达到处理边缘的效果！

```
convert -bordercolor red -border 5x5 input.jpg output.jpg
```

 简单的加上 5 个像素宽的红边！

- convert 顾名思义就是对图像进行转化，它主要用来对图像进行格式的转化，同时还可以缩放、剪切、模糊、反转等操作。

- 格式转化比如把 foo.jpg 转化为 foo.png：

```
convert foo.jpg foo.png
```

 如果要想把目录下所有的 jpg 文件都转化为 gif，我们可借助于 shell 的强功能：find ./ -name "*.jpg" -exec convert {} {}.gif \; 转化后的 gif 名称为 _jpg。

if，这样看起来不太自然，没关系，我们可以再来一步：rename jpg.gif .gif *.jpg.gif 本来，我想在 ind 的时候，用 basename 来取得不带后缀的文件名的，这样就不会形成.jpg.gif 这种丑陋的名子了可是不知道为什么，就是不行，如果你知道的话，告诉我或者，你也可用 shell script 来完成上述的作：

```
for i in *.jpg
do
    convert $i `basename $i .jpg`.gif
done
```

- 我们还可用 mogrify 来完成同样的效果：

```
mogrify -format png *.jpg
```

 命令将会把目录下面所有的 jpg 文件转化为 png 格式。

- convert 还可以把多张照片转化成 pdf 格式：

```
convert *.jpg foo.pdf
```

- 大小缩放比如我们要为一个普通大小的图片做一个缩略图，我们可以这样

```
convert -resize 100x100 foo.jpg thumbnail.jpg
```

- 你也可以用百分比，这样显的更为直观：

`convert -resize 50%x50% foo.jpg thumbnail.jpg`

- 会自动地考虑在缩放图像大小时图像的高宽的比例，也就是说新的图像的高宽比与原图相同我们还可以批量生成缩略图：

`mogrify -sample 80x60 *.jpg`

注意，这个命令会覆盖原来的图片，不过你可以在操作前，先把你的图片备份一下。

- 加边框在一张照片的四周加上边框，可以用 `-mattecolor` 参数，比如某位同志牺牲了，我们需要为做一张黑边框的遗像，可以这样：

`convert -mattecolor "#000000" -frame 60x60 yourname.jpg rememberyou.png` 其中，"#000000"是边框的颜色，边框的大小为 60×60 你也可以这样加边框：

`convert -border 60x60 -bordercolor "#000000" yourname.jpg rememberyou.png` 在图片上加字

`convert -fill green -pointsize 40 -draw 'text 10,50 "charry.org"' foo.png bar.png` 上面的命令在离图片的左上角 10×50 的位置，用绿色的字写下 charry.org，如果你要指定别的字体，可以用 `-font` 参数。

- 模糊高斯模糊：

`convert -blur 80 foo.jpg foo.png-blur` 参数还可以这样 `-blur 80×5`。后面的那个 5 表示的是 Sigma 的值，这个是图像术语，我也不太清楚，总之，它的值对模糊的效果起关键的作用。

- 翻转上下翻转：

`convert -flip foo.png bar.png`

- 左右翻转：

`convert -flop foo.png bar.png`

- 反色形成底片的样子：

`convert -negate foo.png bar.png`

- 单色把图片变为黑白颜色：

`convert -monochrome foo.png bar.png`

- 加噪声

`convert -noise 3 foo.png bar.png`

- 油画效果我们可用这个功能，把一张普通的图片，变成一张油画，效果非常的逼真

`convert -paint 4 foo.png bar.png`

- 旋转把一张图片，旋转一定的角度：

`convert -rotate 30 foo.png bar.png`

上面的 30，表示向右旋转 30 度，如果要向左旋转，度数就是负数。

- 炭笔效果

`convert -charcoal 2 foo.png bar.png` 形成炭笔或者说是铅笔画的效果。

- 散射毛玻璃效果：

`convert -spread 30 foo.png bar.png` 漩涡以图片的中心作为参照，把图片扭转，形成漩涡的效果：

`convert -swirl 67 foo.png bar.png` 凸起效果用 `-raise` 来创建凸边：

`convert -raise 5x5 foo.png bar.png` 执行后，你会看到，照片的四周会一个 5×5 的边，如果你要

个凹下去的边，把-raise 改为+raise 就可以了。其实凸边和凹边看起来区别并不是很大。

其他

其他其他功能都是不太常用的，如果你感兴趣的话，可以看它的联机文档

import 是一个用于屏幕截图的组件，下面列出的是我们常用的功能，其他的功能，你参考它的 man 了。

截取屏幕的任一矩形区域**import foo.png** 在输入上述的命令后，你的鼠标会变成一个十字，这个时候，你只要在想要截取的地方划一个矩形就可以了

截取程序的窗口**import -pause 3 -frame foo.png** 回车后，用鼠标在你想截的窗口上点一下即可。数-frame 的作用是告诉 import，截图的时候把目标窗口的外框架带上，参数 -pause 的作用很重要你可以试着把它去掉，对比一下，你会发现，目标窗口的标题栏是灰色的，pause 就是让 import 稍延迟一下，等你的目标窗口获得焦点了，才开始截图，这样的图才比较自然。

截取一个倾斜的窗口如果想让你的截图比较 cool，你可以把截取一个倾斜的窗口，方法如下：

import -rotate 30 -pause 3 -frame foo.png 截取整个屏幕

import -pause 3 -window root screen.png 注意，暂停了 3 秒钟，你需要在 3 秒钟内切换到需要取的画面噢。

displaydisplay 应该是我们使用的最为频繁的图像处理软件了，毕竟，还是看的多

- 显示图片
 - **display foo.png** 如果你要显示多个文件，你可以使用通配符
 - **display *.png** 幻灯片
 - **display -delay 5 ***每隔 5 个百分之秒显示一张图片
- 一些快捷键
 - **space**(空格): 显示下一张图片
 - **backspace**(回删键):显示上一张图片
 - **h**: 水平翻转
 - **v**: 垂直翻转
 - **/**:顺时针旋转 90 度
 - ****:逆时针旋转 90 度
 - **>**: 放大
 - **<**: 缩小
 - **F7**:模糊图片
 - **Alt+s**:把图片中间的像素旋转
 - **Ctrl+s**:图象另存
 - **Ctrl+d**:删除图片
 - **q**: 退出

find && convert

- 对当前文件夹下图片进行压缩

```
find ./ -regex '.*\.(jpg|JPG|png|PNG|jpeg|)' -size +50k -exec convert -quality 75 {} {} \;
```

- 对格式 260*345 图片在 50Kb 以上的图片，图片质量为 75 进行压缩

```
find ./ -regex '.*\.(jpg|JPG|png|PNG|jpeg|)' -size +50k -exec convert -resize 260x345 -quality 75 {} {} \;
```