

# How to compress JPEG images from the command line on Linux

**Author :** Dan Nanni

**Categories :** [Graphics](#)

**Tagged as :** [cli](#), [compressjpeg](#)

**Question:** I have several JPEG photo images I have taken using a digital camera. I would like to optimize and compress the JPEG pictures before uploading them to Dropbox. What is the easiest way to compress JPEG images without losing their quality on Linux?

Image resolution of today's picture taking gadgets (e.g., smartphones, digital cameras) keeps increasing. Even 36.3 Megapixel Nikon D800 recently hit the consumer market, and this trend will continue. While modern gadgets continue to produce increasingly high resolution images, we may often want to post-process and compress their sizes before uploading them to a storage-limited and bandwidth-restricted cloud.

In fact, there is a way to compress JPEG images easily on Linux. A command-line tool called `jpegoptim` allows you do "lossless" optimization on JPEG images, so you can compress JPEG pictures without sacrificing their quality. In case your storage or bandwidth budget is really low, `jpegoptim` allows you to do "lossy" compression as well by adjusting image quality.

For those interested in compressing PNG images, refer to [this guideline](#) instead.

## Install Jpegoptim on Linux

To install `jpegoptim` on Ubuntu, Debian or Linux Mint:

```
$ sudo apt-get install jpegoptim
```

To install `jpegoptim` on Fedora:

```
$ sudo yum install jpegoptim
```

To install `jpegoptim` on CentOS/RHEL, first enable [EPEL repo](#), and then run:

```
$ sudo yum install jpegoptim
```

## Compress JPEG Images Losslessly

## Ask Xmodulo

Find answers to commonly asked Linux questions

<http://ask.xmodulo.com>

---

To compress a JPG picture losslessly, simply run:

```
$ jpegoptim photo.jpg
```

```
photo.jpg 2048x1536 24bit N ICC JFIF [OK] 882178 --> 821064 bytes (6.93%), optimized.
```

Note that the original input image will be overwritten with a compressed image.

If `jpegoptim` is not able to further optimize an image losslessly, it will skip overwriting it.

```
$ jpegoptim -v photo.jpg
```

```
photo.jpg 2048x1536 24bit N ICC JFIF [OK] 821064 --> 821064 bytes (0.00%), skipped.
```

If you want to preserve an original image, use `-d` option to specify a target directory.

```
$ jpegoptim -d ./compressed photo.jpg
```

An compressed image will then be placed (with the same name as the input file) in `./compressed` directory.

If you want to preserve file creation/modification time, use `-p` option as follows. Then a compressed image will be placed with the same date and time as the original image.

```
$ jpegoptim -d ./compressed -p photo.jpg
```

If you simply check out possible lossless compression ratio without actually compressing it, use `-n` option to "simulate" compression. Then it will simply print results without actually performing compression.

```
$ jpegoptim -n photo.jpg
```

## Compress JPEG Images Lossily

In case you really want to save storage space, you can do lossy compression on large JPEG pictures.

## Ask Xmodulo

Find answers to commonly asked Linux questions

<http://ask.xmodulo.com>

---

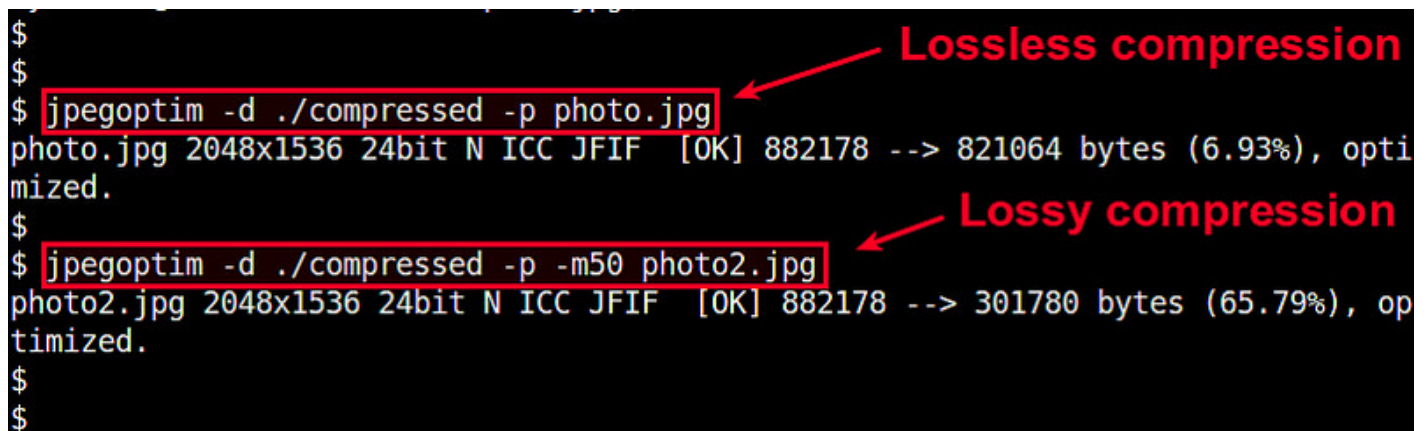
In this case, use "-m" option, where maximum quality is specified in the range of 0 and 100 (0 is the highest quality, and 100 is the lowest quality).

For example, to compress an image with 50% quality:

```
$ jpegoptim -m50 photo.jpg
```

```
photo.jpg 2048x1536 24bit N ICC JFIF [OK] 882178 --> 301780 bytes (65.79%), optimized.
```

You will get a smaller image at the cost of reduced quality.



```
$  
$  
$ jpegoptim -d ./compressed -p photo.jpg  
photo.jpg 2048x1536 24bit N ICC JFIF [OK] 882178 --> 821064 bytes (6.93%), optimized.  
$  
$ jpegoptim -d ./compressed -p -m50 photo2.jpg  
photo2.jpg 2048x1536 24bit N ICC JFIF [OK] 882178 --> 301780 bytes (65.79%), optimized.  
$  
$
```

**Lossless compression**

**Lossy compression**

## Compress Multiple JPEG Images in a Batch

Often times you need to compress many JPEG image files in a directory. In that case, you can use the following shell script.

Alternatively, use [xargs command](#) to achieve the same:

```
$ find . -name "*.jpg" | xargs jpegoptim -d ./compressed -p
```