ImageMagick Image converter and editor

START HERE

`magick <in> <out>`
Convert by extension

`magick <in> -resize 800x <out>`
Resize to width 800

`magick <in> -crop 400x300+10+20 +repage <out>`
Cut out a region

`magick <in> -rotate 90 <out>`
Rotate clockwise

`magick <in> -strip -quality 85 <out>`
Shrink a JPEG

`magick identify <file>`
Show size and format

`magick mogrify -resize 50% *.jpg`
Batch, overwrites originals

`magick montage *.jpg -geometry +4+4 <out>`
Contact sheet

Read input files → **Settings** - background, -gravity → **Operators** -resize, -crop... → **Image list** ( ) stacks, -append → **Write** output name sets format

`settings` `operators` `sequence` `encode`

CONCEPTS

Order matters: settings (-background, -gravity, -quality) must come before the operator that uses them, and options apply to the images read so far.

ImageMagick 7 has one entry point, magick. IM6 used convert, mogrify, identify and montage directly; in IM7 those are subcommands (magick mogrify). Plain convert still works as a legacy alias but is deprecated.

Quote geometry with < > ^ ! and filenames with [ ] so the shell leaves them alone. On Linux, escape stack parentheses as \(and \).

CONVERT & INSPECT

`magick <in> <out>`
Convert; format follows the extension

`magick identify <file>`
One-line size, format, depth

`magick identify -verbose <file>`
Full metadata and statistics

`magick identify -format "%wx%h\n" <files>`
Print just the dimensions
Escapes: %w width, %h height, %m format, %b size, %f name, %[exif:] EXIF*

`magick <fmt>:<in> <out>`
Force input or output format

`magick '<in>[0]' <out>`
Read only the first frame or page

`magick <in> png:-`
Write to stdout

`magick -version`
Version and delegates

RESIZE & CROP

`-resize <WxH>`
Fit inside the box, keep aspect

`-resize <W>x`
Set width, height follows

`-resize <N>%`
Scale by percentage

`-resize 800x600!`
Force exact size, ignore aspect

`-resize 800x600>`
Shrink only if larger

RESIZE & CROP (cont.)

`-resize 800x600^`
Fill the box (minimum size)

`-thumbnail <WxH>`
Resize and drop metadata, fast

`-crop <WxH>+<x>+<y>`
Cut out a region

`-crop <WxH> +repage`
Tile into pieces, reset canvas
Without +repage the saved page offset can leave the output with a larger canvas.

`-gravity center -crop <WxH>+0+0`
Crop around a gravity point

`-trim +repage`
Remove uniform border

`-extent <WxH>`
Pad or crop to a canvas size

`-resize <WxH> -gravity center -extent <WxH>`
Letterbox into a fixed size

ROTATE & ORIENT

`-rotate <deg>`
Rotate clockwise, fills with background

`-auto-orient`
Apply the EXIF orientation

`-flip / -flop`
Mirror vertically / horizontally

`-background <color> -rotate <deg>`
Choose the fill for new corners

`-gravity <dir>`
Anchor for crop, extent, annotate, composite
Directions: NorthWest, North, NorthEast, West, Center, East, SouthWest, South, SouthEast

QUALITY & COLOUR

`-quality <N>`
JPEG/WebP quality 1-100 (PNG: compression)

`-strip`
Remove EXIF, profiles and comments

`-colorspace Gray`
Convert to grayscale

`-colorspace sRGB`
Convert to sRGB

`-interlace Plane`
Progressive JPEG
Also makes interlaced PNG and GIF; -interlace None gives a baseline JPEG.

QUALITY & COLOUR (cont.)

`-density <dpi>`
Set resolution metadata or PDF render DPI
Put -density before the input to rasterise a PDF or SVG at that resolution.

`-normalize`
Stretch contrast to the full range

`-brightness-contrast <b>x<c>`
Adjust brightness and contrast

`-modulate <b>,<s>,<h>`
Brightness, saturation, hue (100 = no change)

`-blur <r>x<sigma>`
Gaussian blur

`-sharpen <r>x<sigma>`
Sharpen

`-unsharp <r>x<s>+<gain>+<thr>`
Unsharp mask, good after downsizing

TRANSPARENCY & COMPOSE

`-alpha set / -alpha off`
Add or drop the alpha channel

`-background <color> -alpha remove`
Flatten transparency onto a colour

`-background <color> -flatten`
Merge layers onto a background

`-transparent <color>`
Make one colour transparent

`magick <base> <over> -composite <out>`
Overlay one image on another

`magick <base> <over> -gravity <dir> -geometry +<x>+<y> -composite <out>`
Place an overlay at an anchor

`-append / +append`
Stack vertically / side by side

`\( <in> -resize <WxH> \)`
Group operations on a sub-list

TEXT & DRAWING

`-gravity South -annotate +0+10 "<text>"`
Draw text at an anchor

`-font <name or file>`
Set the font

`-pointsize <N>`
Set the text size

TEXT & DRAWING (cont.)

`-fill <color>`
Fill colour for text and shapes

`-draw "rectangle x1,y1 x2,y2"`
Draw a rectangle

`-size <WxH> xc:<color> <out>`
Create a plain canvas

`-size <W>x label:"<text>"`
Render text as an image

BATCH & MONTAGE

`magick mogrify -resize <WxH> <files>`
Edit files in place (overwrites)
Test on copies first; mogrify replaces the originals unless -format or -path is used.

`magick mogrify -format jpg *.png`
Convert to new files, keep originals

`magick mogrify -path <dir> -resize <WxH> <files>`
Write results to another folder

`magick montage <files> -geometry +4+4 <out>`
Tile images into a contact sheet

`magick montage -tile <C>x<R> ...`
Grid of columns by rows

`magick montage -label %f ...`
Caption tiles with the file name

`magick <in.pdf> page-%d.png`
One image per PDF page

`magick <files> <out.pdf>`
Combine images into a PDF

RECIPES

Web thumbnails, safe and small

```
# originals stay untouched
mkdir -p out
magick mogrify -path out -auto-orient \
  -thumbnail '400x400>' -strip
-quality 82 *.jpg
```

Watermark in the corner

```
magick <in> \( <logo.png> -resize 20% \) \
  -gravity SouthEast -geometry +16+16 \
  -composite <out>
```

Square avatar from any photo

```
magick <in> -auto-orient -resize 256x256^ \
  -gravity center -extent 256x256 <out>
```