## NAME

montage - creates a composite image by combining several separate images

## SYNOPSIS

**montage [** *options* **...]** *file* [ [ *options* ...] *file* ...] *output_file*

## DESCRIPTION

**montage** creates a composite image by combining several separate images. The images are tiled on the composite image with the name of the image optionally appearing just below the individual tile.

The composite image is constructed in the following manner. First, each image specified on the command line, except for the last, is scaled to fit the maximum tile size. The maximum tile size by default is 120x120. It can be modified with the **-geometry** command line argument or X resource. See **OPTIONS** for more information on command line arguments. See **X(1)** for more information on X resources. Note that the maximum tile size need not be a square. To respect the aspect ratio of each image append ˜ to the geometry specification.

Next the composite image is initialized with the color specified by the **-background** command line argument or X resource. The width and height of the composite image is determined by the title specified, the maximum tile size, the number of tiles per row, the tile border width and height, the image border width, and the label height. The number of tiles per row specifies how many images are to appear in each row of the composite image. The default is to have 5 tiles in each row and 4 tiles in each column of the composite. A specific value is specified with **-tile**. The tile border width and height, and the image border width defaults to the value of the X resource **-borderwidth**. It can be changed with the **-borderwidth** or **-geometry** command line argument or X resource. The label height is determined by the font you specify with the **-font** command line argument or X resource. If you do not specify a font, a font is chosen that allows the name of the image to fit the maximum width of a tiled area. The label colors is determined by the **-background** and **-foreground** command line argument or X resource. Note, that if the background and foreground colors are the same, labels will not appear.

Initially, the composite image title is placed at the top if one is specified (refer to **-foreground** X resource). Next, each image is set onto the composite image, surrounded by its border color, with its name centered just below it. The individual images are left-justified within the width of the tiled area. The order of the images is the same as they appear on the command line unless the images have a scene keyword. If a scene number is specified in each image, then the images are tiled onto the composite in the order of their scene number. Finally, the last argument on the command line is the name assigned to the composite image. By default, the image is written in the **MIFF** format and can be viewed or printed with **display(1)**.

Note, that if the number of tiles exceeds the default number of 20 (5 per row, 4 per column), more than one composite image is created. To ensure a single image is produced, use **-tile** to increase the number of tiles to meet or exceed the number of input images.

Finally, Finally, to create one or more empty spaces in the sequence of tiles, use the **NULL** image format.

## EXAMPLES

To create a montage of a cockatoo, a parrot, and a hummingbird and write it to a file called birds, use:

```
montage cockatoo.miff parrot.miff hummingbird.miff birds.miff
```

To tile several bird images so that they are at most 256 pixels in width and 192 pixels in height, surrounded by a red border, and separated by 10 pixels of background color, use:

```
montage -geometry 256x192+10+10 -bordercolor red birds.* montage.miff
```

To create an unlabeled parrot image, 640 by 480 pixels, and surrounded by a border of black, use:

```
montage -geometry 640x480 -bordercolor black -label "" parrot.miff bird.miff
```

To create an image of an eagle with a textured background, use:

```
montage -texture bumps.jpg eagle.jpg eagle.png
```

To join several GIF images together without any extraneous graphics (e.g. no label, no shadowing, no surrounding tile frame), use:

```
montage +frame +shadow +label -geometry 50x50+0+0 -tile 5x1 *.gif joined.gif
```

## OPTIONS

**-adjoin** join images into a single multi-image file.

**-colors** *value*

preferred number of colors in the image.

The actual number of colors in the image may be less than your request, but never more. Note, this is a color reduction option. Images with less unique colors than specified with this option will have any duplicate or unused colors removed. Refer to **quantize(9)** for more details.

Note, options **-dither**, **-colorspace**, and **-treedepth** affect the color reduction algorithm.

**-colorspace** *value*

the type of colorspace: **GRAY**, **OHTA**, **RGB**, **Transparent**, **XYZ**, **YCbCr**, **YIQ**, **YPbPr**, **YUV**, or **CMYK**.

Color reduction, by default, takes place in the RGB color space. Empirical evidence suggests that distances in color spaces such as YUV or YIQ correspond to perceptual color differences more closely than do distances in RGB space. These color spaces may give better results when color reducing an image. Refer to **quantize(9)** for more details.

The **Transparent** color space behaves uniquely in that it preserves the matte channel of the image if it exists.

The **-colors** or **-monochrome** option is required for this option to take effect.

**-comment** *string*

annotate an image with a comment.

By default, each image is commented with its file name. Use this option to assign a specific comment to the image. Optionally you can include the image filename, type, width, height, or scene number by embedding special format characters. Embed **%f** for filename, **%d** for directory, **%e** for filename extention, **%t** for top of filename, **%m** for magick, **%w** for width, **%h** for height, **%s** for scene number, **%b** for file size, or **\n** for newline. For example,

```
-comment "%m:%f %wx%h"
```

produces an image comment of **MIFF:bird.miff 512x480** for an image titled **bird.miff** and whose width is 512 and height is 480.

If the first character of *string* is @, the image comment is read from a file titled by the remaining characters in the string.

**-compose** *operator*

the type of image composition.

By default, each of the composite image pixels are replaced by the corresponding image tile pixel. You can choose an alternate composite operation:

Over
In
Out
Atop
Xor
Plus
Minus
Add
Subtract
Difference
Bumpmap
Replace
ReplaceRed
ReplaceGreen
ReplaceBlue
ReplaceMatte

The operations behaves as follows:

**Over** The result will be the union of the two image shapes, with *composite image* obscuring *image* in the region of overlap.

**In** The result is simply *composite image* cut by the shape of *composite image window*. None of the image data of *image* will be in the result.

**Out** The resulting image is *composite image* with the shape of *image* cut out.

**Atop** The result is the same shape as image *image*, with *composite image* obscuring *image* where the image shapes overlap. Note this differs from **over** because the portion of *composite image* outside *image*'s shape does not appear in the result.

**Xor** The result is the image data from both *composite image* and *image* that is outside the overlap region. The overlap region will be blank.

**Plus** The result is just the sum of the image data. Output values are cropped to 255 (no overflow). This operation is independent of the matte channels.

**Minus** The result of *composite image* − *image*, with underflow cropped to zero. The matte channel is ignored (set to 255, full coverage).

**Add** The result of *composite image* + *image*, with overflow wrapping around (*mod* 256).

**Subtract**
The result of *composite image* - *image*, with underflow wrapping around (*mod* 256). The **add** and **subtract** operators can be used to perform reversible transformations.

**Difference**
The result of abs(*composite image* − *image*). This is useful for comparing two very similar images.

**Bumpmap**
The result of *image* shaded by *composite image*.

**Replace**
The resulting image is *image* replaced with *composite image*. Here the matte information is

ignored.

**ReplaceRed**
The resulting image is the red layer in *image* replaced with the red layer in *composite image*. The other layers are copied untouched.

**ReplaceGreen**
The resulting image is the green layer in *image* replaced with the green layer in *composite image*. The other layers are copied untouched.

**ReplaceBlue**
The resulting image is the blue layer in *image* replaced with the blue layer in *composite image*. The other layers are copied untouched.

**ReplaceMatte**
The resulting image is the matte layer in *image* replaced with the matte layer in *composite image*. The other layers are copied untouched.

The image compositor requires a matte, or alpha channel in the image for some operations. This extra channel usually defines a mask which represents a sort of a cookie-cutter for the image. This is the case when matte is 255 (full coverage) for pixels inside the shape, zero outside, and between zero and 255 on the boundary. If *image* does not have a matte channel, it is initialized with 0 for any pixel matching in color to pixel location (0,0), otherwise 255 (to work properly **borderwidth** must be 0).

**-compress** *type*
the type of image compression: *None*, *JPEG*, *LZW*, *RunlengthEncoded*, or *Zip*.

Specify **+compress** to store the binary image in an uncompressed format. The default is the compression type of the specified image file.

**-crop** *<width>{%}x<height>{%}{+-}<x offset>{+-}<y offset>*
preferred size and location of the cropped image. See **X(1)** for details about the geometry specification.

To specify a percentage width or height instead, append **%**. For example to crop the image by ten percent on all sides of the image, use **-crop 10%**.

Use cropping to tile only a particular area of an image. Use **-crop 0x0** to remove edges that are the background color.

The equivalent X resource for this option is **cropGeometry** (class **CropGeometry**). See **X RESOURCES** for details.

**-density** *<width>x<height>*
vertical and horizontal resolution in pixels of the image.

This option specifies an image density when decoding a Postscript or Portable Document page. The default is 72 pixels per inch in the horizontal and vertical direction. This option is used in concert with **-page**.

**-display** *host:display[.screen]*
specifies the X server to contact; see **X(1)**.

Specify **+display** if an X server is not available. The label font is obtained from the X server. If none is available, the composite image will not have labels. Since the X server is necessary to read X resources, all options must be set via the command line when +display is specified.

**-dispose** *method*

GIF disposal method.

Here are the valid methods:

0 No disposal specified.
1 Do not dispose.
2 Restore to background color.
3 Restore to previous.

**-dither** apply Floyd/Steinberg error diffusion to the image.

The basic strategy of dithering is to trade intensity resolution for spatial resolution by averaging the intensities of several neighboring pixels. Images which suffer from severe contouring when reducing colors can be improved with this option.

The **-colors** or **-monochrome** option is required for this option to take effect.

**-draw** *string*

annotate an image with one or more graphic primitives.

Use this option to annotate an image with one or more graphic primitives. The primitives include

rectangle
circle
polygon
color
matte
text
image

**Rectangle**, **color**, **matte**, **text**, and **image** require an upper left and lower right coordinate. **Circle** requires the center coordinate and a coordinate on the outer edge. Finally, **polygon** requires three or more coordinates defining its boundaries. Coordinates are integers separated by an optional comma. For example, to define a circle centered at 100,100 that extends to 150,150 use:

-draw 'circle 100,100 150,150'

Use **color** to change the color of a pixel. Follow the pixel coordinate with a method:

point
replace
floodfill
reset

Consider the target pixel as that specified by your coordinate. The **point** method recolors the target pixel. The **replace** method recolors any pixel that matches the color of the target pixel. **Floodfill** recolors any pixel that matches the color of the target pixel and is a neighbor. Finally, **reset** recolors all pixels.

Use **matte** to the change the pixel matte value to transparent. Follow the pixel coordinate with a method (see the **color** primitive for a description of methods). The **point** method changes the matte value of the target pixel. The **replace** method changes the matte value of any pixel that matches the color of the target pixel. **Floodfill** changes the matte value of any pixel that matches the color of the target pixel and is a neighbor. Finally **reset** changes the matte value of all pixels.

Use **text** to annotate an image with text. Follow the text coordinates with a string. If the string has embedded spaces, enclose it in double quotes. Optionally you can include the image filename, type, width, height, or scene number by embedding special format characters. Embed **%f** for filename, **%d** for directory, **%e** for filename extention, **%t** for top of filename, **%m** for magick, **%w** for width, **%h** for height, **%s** for scene number, **%b** for file size, or **\n** for newline. For example,

    -draw 'text 100,100 "%m:%f %wx%h"'

annotates the image with **MIFF:bird.miff 512x480** for an image titled **bird.miff** and whose width is 512 and height is 480.

If the first character of the string is @, the text is read from a file titled by the remaining characters in the string.

Use **image** to composite an image with another image. Follow the image coordinates with the filename of an image.

If the first character of *string* is @, the text is read from a file titled by the remaining characters in the string.

You can set the primitive color, font color, and font bounding box color with **-pen**, **-font**, and **-box** respectively. Options are processed in command line order so be sure to use **-pen** *before* the **-draw** option.

**-font** *name*
use this font when annotating the image with text.

If the font is a fully qualified X server font name, the font is obtained from an X server (e.g. -*-helvetica-medium-r-*-*-12-*-*-*-*-*-iso8859-*). To use a TrueType font, precede the TrueType filename with a @ (e.g. @times.ttf). Otherwise, specify a Postscript font (e.g. helvetica).

**-frame** surround the image with an ornamental border.

The color of the border is specified with the **-mattecolor** command line option. If no frame is desired, use **+frame**.

**-gamma** *value*
level of gamma correction.

The same color image displayed on two different workstations may look different due to differences in the display monitor. Use gamma correction to adjust for this color difference. Reasonable values extend from 0.8 to 2.3.

You can apply separate gamma values to the red, green, and blue channels of the image with a gamma value list delineated with slashes (i.e. 1.7/2.3/1.2).

Use **+gamma** to set the image gamma level without actually adjusting the image pixels. This option is useful if the image is of a known gamma but not set as an image attribute (e.g. PNG images).

**-geometry** *<width>x<height>+<border width>+<border height>{!}{<}{>}*
preferred tile and border size of each tile of the composite image.

By default, the width and height are maximum values. That is, the image is expanded or contracted to fit the width and height value while maintaining the aspect ratio of the image. Append an exclamation point to the geometry to force the image size to exactly the size you specify. For example, if you specify **640x480!** the image width is set to 640 pixels and height to 480. If only one factor is specified, both the width and height assume the value.

Use **>** to change the dimensions of the image *only* if its size exceeds the geometry specification. **<** resizes the image *only* if its dimensions is less than the geometry specification. For example, if you specify **640x480>** and the image size is 512x512, the image size does not change. However, if the image is 1024x1024, it is resized to 640x480.

Each image is surrounded by a border whose size in pixels is specified as *<border width>* and *<border height>* and whose color is the background color. By default, the tile size is 256x256 and there is no border.

The equivalent X resource for this option is **imageGeometry** (class **ImageGeometry**). See **X RESOURCES** for details.

**-gravity** *direction*

direction image gravitates to within a tile. See **X(1)** for details about the gravity specification.

A tile of the composite image is a fixed width and height. However, the image within the tile may not fill it completely (see **-geometry**). The direction you choose specifies where to position the image within the tile. For example *Center* gravity forces the image to be centered within the tile. By default, the image gravity is *Center*.

**-interlace** *type*

the type of interlacing scheme: **None**, **Line**, **Plane**, or **Partition**. The default is **None**.

This option is used to specify the type of interlacing scheme for raw image formats such as **RGB** or **YUV**. **No** means do not interlace (RGBRGBRGBRGBRGBRGB...), **Line** uses scanline interlacing (RRR...GGG...BBB...RRR...GGG...BBB...), and **Plane** uses plane interlacing (RRRRRR...GGGGGG...BBBBBB...). **Partition** is like plane except the different planes are saved to individual files (e.g. image.R, image.G, and image.B).

Use **Line**, or **Plane** to create an interlaced GIF or progressive JPEG image.

**-label** *name*

assign a label to an image.

By default, each image is labeled with its file name. Use this option to assign a specific label to the image. Optionally you can include the image filename, type, width, height, or scene number in the label by embedding special format characters. Embed **%f** for filename, **%d** for directory, **%e** for filename extention, **%t** for top of filename, **%m** for magick, **%w** for width, **%h** for height, **%s** for scene number, **%b** for file size, or **\n** for newline. For example,

```
-label "%m:%f %wx%h"
```

produces an image label of **MIFF:bird.miff 512x480** for an image titled **bird.miff** and whose width is 512 and height is 480.

If the first character of *string* is @, the image label is read from a file titled by the remaining characters in the string.

**-matte** store matte channel if the image has one.

**-mode** *type*

the type of montage: **Frame**, **Unframe**, **Concatentate**. The default is **Unframe**.

This option is for convenience. You can obtain the desired result by setting individual options (e.g. **Unframe** is equivalent to **+frame +shadow +borderwidth**).

**-monochrome**
transform the image to black and white.

**-page** *<width>{%}x<height>{%}{+-}<x offset>{+-}<y offset>{!}{<}{>}*
preferred size and location of an image canvas.

Use this option to specify the dimensions of the Postscript page in dots per inch or a TEXT page in pixels. The choices for a Postscript page are:

| | |
|---|---|
| Letter | 612x 792 |
| Tabloid | 792x1224 |
| Ledger | 1224x 792 |
| Legal | 612x1008 |
| Statement | 396x 612 |
| Executive | 540x 720 |
| A3 | 842x1190 |
| A4 | 595x 842 |
| A5 | 420x 595 |
| B4 | 729x1032 |
| B5 | 516x 729 |
| Folio | 612x 936 |
| Quarto | 610x 780 |
| 10x14 | 720x1008 |

For convenience you can specify the page size by media (e.g. A4, Ledger, etc.). Otherwise, **-page** behaves much like **-geometry** (e.g. -page letter+43+43>).

To position a GIF image, use -page *{+-}<x offset>{+-}<y offset>* (e.g. -page +100+200). For a Postscript page, the image is sized as in **-geometry** and positioned relative to the lower left hand corner of the page by *{+-}<x offset>{+-}<y offset>*. The default is to center the image within the page. If the image size exceeds the Postscript page, it is reduced to fit the page.

The default page dimensions for a TEXT image is 612x792.

This option is used in concert with **-density**.

**-pen** *color*
set the color of the font.

See **X(1)** for details about the color specification.

**-pointsize** *value*
pointsize of the Postscript font.

**-quality** *value*
JPEG/MIFF/PNG compression level.

For the JPEG image format, quality is 0 (worst) to 100 (best). The default quality is 75.

Quality for the MIFF and PNG image format sets the amount of image compression (quality / 10) and filter-type (quality % 10). Compression quality values range from 0 (worst) to 100 (best). If filter-type is 4 or less, the specified filter-type is used for all scanlines:

0: none
1: sub
2: up
3: average
4: Paeth

If filter-type is 5, adaptive filtering is used when quality is greater than 50 and the image does not have a color map, otherwise no filtering is used.

If filter-type is 6 or more, adaptive filtering with *minimum-sum-of-absolute-values* is used.

The default is quality is 75. Which means nearly the best compression with adaptive filtering.

For further information, see the PNG specification (RFC 2083), <http://www.w3.org/pub/WWW/TR>.

**-rotate** *degrees{<}{>}*
apply Paeth image rotation to the image.

Use **>** to rotate the image *only* if its width exceeds the height. **<** rotates the image *only* if its width is less than the height. For example, if you specify **-90>** and the image size is 480x640, the image is not rotated by the specified angle. However, if the image is 640x480, it is rotated by -90 degrees.

Empty triangles left over from rotating the image are filled with the color defined as **bordercolor** (class **borderColor**).

**-scene** *value*
image scene number.

**-shadow**
add a shadow beneath a tile to simulate depth.

**-size** *<width>{%}x<height>{%}{+offset}{!}*
width and height of the image.

Use this option to specify the width and height of raw images whose dimensions are unknown such as **GRAY**, **RGB**, or **CMYK**. In addition to width and height, use **-size** to skip any header information in the image or tell the number of colors in a **MAP** image file, (e.g. -size 640x512+256).

**-texture** *filename*
name of texture to tile onto the image background.

**-tile** *<width>x<height>*
specifies how many tiles are to appear in each row and column of the composite image.

Specify the number of tiles per row with width and tiles per column with height. For example if you want 1 tile in each row and a maximum of 10 tiles in the composite image, use -tile 1x10. The default is to have 5 tiles in each row and 4 tiles in each column of the composite.

**-transparency** *color*
make this color transparent within the image.

**-treedepth** *value*
Normally, this integer value is zero or one. A zero or one tells **montage** to choose a optimal tree

depth for the color reduction algorithm.

An optimal depth generally allows the best representation of the source image with the fastest computational speed and the least amount of memory. However, the default depth is inappropriate for some images. To assure the best representation, try values between 2 and 8 for this parameter. Refer to **quantize(9)** for more details.

The **-colors** or **-monochrome** option is required for this option to take effect.

**-verbose**

print detailed information about the image.

This information is printed: image scene number; image name; image size; the image class (*DirectClass* or *PseudoClass*); the total number of unique colors; and the number of seconds to read and write the image.

In addition to those listed above, you can specify these standard X resources as command line options: -background, -bordercolor, -borderwidth, -font, -foreground, -mattecolor, or -title. See **X RESOURCES** for details.

Options are processed in command line order. Any option you specify on the command line remains in effect until it is explicitly changed by specifying the option again with a different effect. For example, to montage two images, the first with 32 colors and the second with only 16 colors, use:

```
montage -colors 32 cockatoo.1 -colors 16 cockatoo.2 cockatoo.miff
```

By default, the image format is determined by its magic number. To specify a particular image format, precede the filename with an image format name and a colon (i.e. ps:image) or specify the image type as the filename suffix (i.e. image.ps). See **convert(1)** for a list of valid image formats.

When you specify **X** as your image type, the filename has special meaning. It specifies an X window by id, name, or **root**. If no filename is specified, the window is selected by clicking the mouse in the desired window.

Specify *input_file* as - for standard input, *output_file* as - for standard output. If *input_file* has the extension **.Z** or **.gz**, the file is uncompressed with **uncompress** or **gunzip** respectively. If *output_file* has the extension **.Z** or **.gz**, the file size is compressed using with **compress** or **gzip** respectively. Finally, precede the image file name with **|** to pipe to or from a system command.

Use an optional index enclosed in brackets after a file name to specify a desired subimage of a multi-resolution image format like Photo CD (e.g. img0001.pcd[4]) or a range for MPEG images (e.g. video.mpg[50-75]). A subimage specification can be disjoint (e.g. image.tiff[2,7,4]). For raw images, specify a subimage with a geometry (e.g. -size 640x512 image.rgb[320x256+50+50]).

Single images are written with the filename you specify. However, multi-part images (e.g. a multi-page Postscript document with **+adjoin** specified) are written with the filename followed by a period (**.**) and the scene number. You can change this behavior by embedding a **printf** format specification in the file name. For example,

```
image%02d.miff
```

montages files image00.miff, image01.miff, etc.

Note, a composite MIFF image displayed to an X server with **display** behaves differently than other images. You can think of the composite as a visual image directory. Choose a particular tile of the composite and press a button to display it. See **display(1)** and **miff(5)** for details.

## X RESOURCES

**montage** options can appear on the command line or in your X resource file. Options on the command line supersede values specified in your X resource file. See **X(1)** for more information on X resources.

All **montage** options have a corresponding X resource. In addition, **montage** uses the following X resources:

**background (**class **Background)**
Specifies the preferred color to use for the composite image background. The default is #ccc.

**borderColor (**class **BorderColor)**
Specifies the preferred color to use for the composite image border. The default is #ccc.

**borderWidth (**class **BorderWidth)**
Specifies the width in pixels of the composite image border. The default is 2.

**font (**class **Font)**
Specifies the name of the preferred font to use when displaying text within the composite image. The default is 9x15, fixed, or 5x8 determined by the composite image size.

**foreground (**class **Foreground)**
Specifies the preferred color to use for text within the composite image. The default is black.

**matteColor (**class **MatteColor)**
Specify the color of an image frame. A 3D effect is achieved by using highlight and shadow colors derived from this color. The default value is #ccc.

**title (**class **Title)**
This resource specifies the title to be placed at the top of the composite image. The default is not to place a title at the top of the composite image.

## ENVIRONMENT

**display** To get the default host, display number, and screen.

## SEE ALSO

**display(1), animate(1), import(1), mogrify(1), convert(1), combine(1), xtp(1)**

## COPYRIGHT

**ACKNOWLEDGEMENTS**

The MIT X Consortium for making network transparent graphics a reality.

Michael Halle, Spatial Imaging Group at MIT, for the initial implementation of Alan Paeth's image rotation algorithm.

David Pensak, E. I. du Pont de Nemours and Company, for providing a computing environment that made this program possible.

Paul Raveling, USC Information Sciences Institute, for the original idea of using space subdivision for the color reduction algorithm.

**AUTHORS**

John Cristy, E.I. du Pont de Nemours and Company Incorporated