

NAME

ppmlabel – add text to a portable pixmap

SYNOPSIS

ppmlabel [**–angle** *angle*] [**–background transparent** | *colour*] [**–colour** *colour*] [**–file** *filename*] [**–size** *textsize*] [**–text** '*text string*'] [**–x** *column*] [**–y** *row*] ... [*ppmfile*]

DESCRIPTION

ppmlabel uses the text drawing facilities of **ppmdraw** to add text to a portable pixmap. The location, size, baseline angle, colour of the text and background colour (if any) are controlled by command line arguments. The text can be specified on the command line or read from files. Any number of separate text strings can be added by one invocation of **ppmlabel**, limited only by the maximum length of the command line.

If no *ppmfile* is specified, **ppmdraw** reads its input pixmap from standard input.

OPTIONS

The arguments on the **ppmlabel** command line are not options in the strict sense; they are commands which control the placement and appearance of the text being added to the input pixmap. They are executed left to right, and any number of arguments may appear.

All flags can be abbreviated to their shortest unique prefix.

–angle *angle*

Sets the angle of the baseline of subsequent text. *angle* is specified as an integral number of degrees, measured counterclockwise from the row axis of the pixmap.

–background transparent | *colour*

If the argument is “transparent”, text is drawn over the existing pixels in the pixmap. If a *colour* is given (see the **–colour** switch below for information on how to specify colours), rectangles enclosing subsequent text are filled with that colour.

–colour *colour*

Sets the colour for subsequent text. The *colour* can be specified in five ways:

- A name, assuming that a pointer to an X11-style colour names file was compiled in.
- An X11-style hexadecimal specifier: *rgb:r/g/b*, where *r* *g* and *b* are each 1- to 4-digit hexadecimal numbers.
- An X11-style decimal specifier: *rgbi:r/g/b*, where *r* *g* and *b* are floating point numbers between 0 and 1.
- For backwards compatibility, an old-X11-style hexadecimal number: *#rgb*, *#rrggbb*, *#rrrgggbbb*, or *#rrrrggggbbbb*.
- For backwards compatibility, a triplet of numbers separated by commas: *r,g,b*, where *r* *g* and *b* are floating point numbers between 0 and 1. (This style was added before MIT came up with the similar *rgbi* style.)

–file *filename*

Reads text from the file *filename* and draws it on successive lines.

–size *textsize*

Sets the height of the tallest characters above the baseline to *textsize* pixels.

–text '*text string*'

Draws the given text string (which must be quoted if it contains spaces). The location for subsequent text is advanced by 1.75 times the current *textsize*, which allows drawing multiple lines of text in a reasonable manner without specifying the position of each line.

–x *column* Sets the column at which subsequent text will be left justified. Depending on the shape of the first character, the actual text may begin a few pixels to the right of this point.

-y row Sets the row which will form the baseline of subsequent text. Characters with descenders, such as “y”, will extend below this line.

BUGS

Text strings are restricted to 7 bit ASCII. The text font used by **ppmdraw** doesn't include definitions for 8 bit ISO 8859/1 characters.

When drawing multiple lines of text with a non-transparent background, it should probably fill the space between the lines with the background colour. This is tricky to get right when the text is rotated to a non-orthogonal angle.

The **-size**, **-x**, and **-y** options **MUST** precede the **-text** option specifying the string they apply to, or they will be silently ignored in favor of the defaults.

SEE ALSO

ppmmake(1), **ppm(5)**

AUTHOR

Copyright (C) 1995 by John Walker (kelvin@fourmilab.ch)

WWW home page: <http://www.fourmilab.ch/>

Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, without any conditions or restrictions. This software is provided “as is” without express or implied warranty.