

NAME

pnmnorm - normalize the contrast in a Netbpm image

SYNOPSIS

pnmnorm [-bpercent *N* | -bvalue *N*] [-wpercent *N* | -wvalue *N*] [-keephues] [-brightmax]

[*ppmfile*]

All options can be abbreviated to their shortest unique prefix. You may use two hyphens instead of one to designate an option. You may use either white space or an equals sign between an option name and its value.

DESCRIPTION

Reads a PNM image (PBM, PGM, or PPM). Normalizes the contrast by forcing the lightest pixels to white, the darkest pixels to black, and linearly rescaling the ones in between; and produces the same kind of file as output. This is pretty useless for a PBM image.

The program first determines a mapping of old brightness to new brightness. For each possible brightness of a pixel, the program determines a corresponding brightness for the output image.

Then for each pixel in the image, the program computes a color which has the desired output brightness and puts that in the output. With a color image, it is not always possible to compute such a color and retain any semblance of the original hue, so the brightest and dimmest pixels may only approximate the desired brightness.

Note that for a PPM image, this is different from separately normalizing the individual color components.

OPTIONS

By default, the darkest 2 percent of all pixels are mapped to black, and the lightest 1 percent are mapped to white. You can override these percentages by using the **-bpercent** and **-wpercent** flags, or you can specify the exact pixel values to be mapped by using the **-bvalue** and **-wvalue** flags. Appropriate numbers for the flags can be gotten from the *ppmhist* tool. If you just want to enhance the contrast, then choose values at elbows in the histogram; e.g. if value 29 represents 3% of the image but value 30 represents 20%, choose 30 for *bvalue*. If you want to lighten the image, then set *bvalue* to 0 and just fiddle with *wvalue*; similarly, to darken the image, set *wvalue* to maxval and play with *bvalue*.

The **-keephues** option says to keep each pixel the same hue as it is in the input; just adjust its intensity. By default, **pnmnorm** normalizes contrast in each component independently (except that the meaning of the **-wpercent** and **-bpercent** options are based on the overall intensities of the colors, not each component taken separately). So if you have a color which is intensely red but dimly green, **pnmnorm** would make the red more intense and the green less intense, so you end up with a different hue than you started with.

If you specify **-keephues**, **pnmnorm** would likely leave this pixel alone, since its overall intensity is medium.

-keephues can cause clipping, because a certain color may be below a target intensity while one of its components is saturated. Where that's the case, **pnmnorm** uses the maximum representable intensity for the saturated component and the pixel ends up with less overall intensity, and a different hue, than it is supposed to have.

This option is meaningless on grayscale images.

Before March 2002, there was no **-keephues** option.

The **-brightmax** option says to use the intensity of the most intense RGB component of a pixel as the

pixel's brightness. By default, **pnmnorm** uses the luminosity of the color as its brightness.

This option is meaningless on grayscale images.

Before March 2002, there was no **-brightmax** option.

SEE ALSO

ppmhist(1), pgmhist(1), pnmgamma(1), ppmbrighten(1), ppmdim(1), pnm(5)