Pnmremap User Manual

pnmremap

Updated: 01 January 2002
Table Of Contents

NAME

pnmremap - replace colors in a PNM image with colors from another set

SYNOPSIS

pnmremap **-mapfile=***palettefile* [**-floyd**|**-fs**|**-nfloyd**|**-nofs**] [**-firstisdefault**] [**-verbose**] [**-missingcolor=***color*] [*pnmfile*]

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

This program is part of Netpbm.

pnmremap replaces the colors in an input image with those from a palette you specify. Where colors in the input are present in the palette, they just stay the same in the output. But where the input contains a color that is not in the palette, **pnmremap** gives you three choices: 1) choose the closest color from the palette; 2) choose the first color from the palette; 3) use a color specified by a command option.

You can also dither, which means rather than mapping pixel by pixel, **pnmremap** uses colors from the palette to try to make multi-pixel regions of the output have the same average color as the input (for another kind of dithering, see **ppmdither**).

Two reasons to use this program are: 1) you want to reduce the number of colors in the input image; and 2) you need to feed the image to something that can handle only certain colors.

To reduce colors, you can generate the palette with **pnmcolormap**.

By default, **pnmremap** maps an input color that is not in the palette to the closest color that is in the palette. Closest means with the smallest cartesian distance in the red, green, blue brightness space (smallest sum of the squares of the differences in red, greeN, and blue ITU-R Recommedation BT.709 gamma-adjusted intensities).

You can instead specify a single default color for **pnmremap** to use for any color in the input image that is not in the palette. Use the **-missing** option for this.

You can also specify that the first color in the palette image is the default. Use the **-firstisdefault** option for this.

The palette is simply a PNM image. The colors of the pixels in the image are the colors in the palette. Where the pixels appear in the image, and the dimensions of the image, are irrelevant. Multiple pixels of the same color are fine. However, a palette image is typically a single row with one pixel per color.

If you specify **-missing**, the color you so specify is in the palette in addition to whatever is in the palette image.

For historical reasons, Netpbm sometimes calls the palette a "colormap." But it doesn't really map anything. **pnmremap** creates its own map, based on the palette, to map colors from the input image to output

colors.

Palette/Image Type Mismatch

In the simple case, the palette image is of the same depth (number of planes, i.e. number of components in each tuple (pixel)) as the input image and **pnmremap** just does a straightforward search of the palette for each input tuple (pixel). In fact, **pnmremap** doesn't even care if the image is a visual image.

But what about when the depths differ? In that case, **pnmremap** converts the input image (in its own memory) to match the palette and then proceeds as above.

There are only two such cases in which **pnmremap** knows how to do the conversion: when one of them is tuple type RGB, depth 3, and the other is tuple type GRAYSCALE or BLACKANDWHITE, depth 1; and vice versa.

In any other case, **pnmremap** issues and error message and fails.

Note that as long as your input and palette images are PNM, they'll always fall into one of the cases **pnmremap** can handle. There's an issue only if you're using some exotic PAM image.

Before Netpbm 10.27 (March 2005), **pnmremap** could not handle the case of a palette of greater depth than the input image. (It would issue an error message and fail in that case).

In any case, the output image has the same tuple type and depth as the palette image.

Multiple Image Stream

pnmremap handles a multiple image input stream, producing a multiple image output stream. The input images need not be similar in any way.

Before Netpbm 10.30 (October 2005), **pnmremap** ignored any image after the first.

Examples

```
pnmcolormap testimg.ppm 256 >palette.ppm
pnmremap -map=palette.ppm testimg.ppm >reduced_testimg.ppm
```

To limit colors to a certain set, a typical example is to create an image for posting on the World Wide Web, where different browsers know different colors. But all browsers are supposed to know the 216 "web safe" colors which are essentially all the colors you can represent in a PPM image with a maxval of 5. So you can do this:

```
pamseq 3 5 >websafe.pam
pnmremap -map=websafe.pam testimg.ppm >websafe_testimg.ppm
```

Another useful palette is one for the 8 color IBM TTL color set, which you can create with

```
pamseq 3 1 >ibmttl.pam
```

If you want to quantize one image to use the colors in another one, just use the second one as the palette. You don't have to reduce it down to only one pixel of each color, just use it as is.

The output image has the same type and maxval as the palette image.

PARAMETERS

There is one parameter, which is required: The file specification of the input PNM file.

OPTIONS

-mapfile=*palettefilename*This names the file that contains the palette image.

This option is mandatory.

-floyd

-fs

-nofloyd

-nofs These options determine whether Floyd-Steinberg dithering is done. Without Floyd-Steinberg, the selection of output color of a pixel is based on the color of only the corresponding input pixel. With Floyd-Steinberg, multiple input pixels are considered so that the average color of an area tends to stay more the same than without Floyd-Steinberg. For example, if you map an image with a black, gray, gray, and white pixel adjacent, to a palette that contains only black and white, it might result in an output of black, black, white, white. Pixel-by-pixel mapping would instead map both the gray pixels to the same color.

Floyd-Steinberg gives vastly better results on images where unmodified quantization has banding or other artifacts, especially when going to a small number of colors such as the above IBM set. However, it does take substantially more CPU time.

-fs is a synomym for **-floyd**. **-nofs** is a synonym for **-nofloyd**.

The default is **-nofloyd**.

-firstisdefault This tells **pnmremap** to map any input color that is not in the palette to the first color in the palette (the color of the pixel in the top left corner of the palette image)

See DESCRIPTION.

If you specify **-firstisdefault**, the maxval of your input must match the maxval of your palette image.

-missingcolor=*color*This specifies the default color for **pnmremap** to map to a color in the input image that isn't in the palette. *color* may or may not be in the palette image; it is part of the palette regardless.

If you specify **-missingcolor**, the maxval of your input must match the maxval of your palette image.

-verbose Display helpful messages about the mapping process.

SEE ALSO

pnmcolormap, **pamseq**, **pnmquant**, **ppmquantall**, **pamdepth**, **ppmdither**, **ppmquant**, **ppm**

AUTHOR

Copyright (C) 1989, 1991 by Jef Poskanzer.

Table Of Contents

SYNOPSIS

DESCRIPTION

Palette/Image Type Mismatch

Multiple Image Stream

Examples

PARAMETERS

OPTIONS

SEE ALSO

AUTHOR