Pnmconvol User Manual

pnmconvol

Updated: 29 June 2005
Table Of Contents

NAME

pnmconvol - general MxN convolution on a PNM image

SYNOPSIS

pnmconvol *convolution_matrix_file* [-nooffset] [*pnmfile*]

Minimum unique abbreviation of option is acceptable. You may use double hyphens instead of single hyphen to denote options. You may use white space in place of the equals sign to separate an option name from its value.

DESCRIPTION

This program is part of Netpbm.

pnmconvol reads two PNM images as input, convolves the second using the first, and writes a PNM image as output.

Convolution means replacing each pixel with a weighted average of the nearby pixels. The weights and the area to average are determined by the convolution matrix (sometimes called a convolution kernel), which you supply by way of the PNM image in the file you identify with the *convolution_matrix_file* argument. There are two ways **pnmconvol** interprets the PNM convolution matrix image pixels as weights: with offsets, and without offsets.

The simpler of the two is without offsets. That is what happens when you specify the **-nooffset** option. In that case, **pnmconvol** simply normalizes the sample values in the PNM image by dividing by the maxval.

For example, here is a sample convolution file that causes an output pixel to be a simple average of its corresponding input pixel and its 8 neighbors, resulting in a smoothed image:

```
P2
3 3
18
2 2 2
2 2 2
2 2 2
```

(Note that the above text is an actual PGM file -- you can cut and paste it. If you're not familiar with the plain PGM format, see the PGM format specification).

pnmconvol divides each of the sample values (2) by the maxval (18) so the weight of each of the 9 input pixels gets is 1/9, which is exactly what you want to keep the overall brightness of the image the same. **pnmconvol** creates an output pixel by multiplying the values of each of 9 pixels by 1/9 and adding.

Note that with maxval 18, the range of possible values is 0 to 18. After scaling, the range is 0 to 1.

For a normal convolution, where you're neither adding nor subtracting total value from the image, but merely moving it around, you'll want to make sure that all the scaled values in (each plane of) your

convolution PNM add up to 1, which means all the actual sample values add up to the maxval.

When you don't specify **-nooffset**, **pnmconvol** applies an offset, the purpose of which is to allow you to indicate negative weights even though PNM sample values are never negative. In this case, **pnmconvol** subtracts half the maxval from each sample and then normalizes by dividing by half the maxval. So to get the same result as we did above with **-nooffset**, the convolution matrix PNM image would have to look like this:

```
P2
3 3
18
10 10 10
10 10 10
10 10 10
```

To see how this works, do the above-mentioned offset: 10 - 18/2 gives 1. The normalization step divides by 18/2 = 9, which makes it 1/9 - exactly what you want. The equivalent matrix for 5x5 smoothing would have maxval 50 and be filled with 26.

Note that with maxval 18, the range of possible values is 0 to 18. After offset, that's -9 to 9, and after normalizing, the range is -1 to 1.

For a normal convolution, where you're neither adding nor subtracting total value from the image, but merely moving it around, you'll want to make sure that all the offset, scaled values in (each plane of) your convolution PNM add up to 1. That means the actual sample values, less half the maxval, add up to half the maxval as in the example above.

The convolution file will usually be a PGM, so that the same convolution gets applied to each color component. However, if you want to use a PPM and do a different convolution to different colors, you can certainly do that.

At the edges of the convolved image, where the convolution matrix would extend over the edge of the image, **pnmconvol** just copies the input pixels directly to the output.

The convolution computation can result in a value which is outside the range representable in the output. When that happens, **pnmconvol** just clips the output, which means brightness is not conserved.

HISTORY

The **-nooffset** option was new in Netpbm 10.23 (July 2004).

SEE ALSO

pnmsmooth, **pgmmorphconv**, **pnmnlfilt**, **pgmkernel**, **pamgauss**, **pnm**

AUTHORS

Copyright (C) 1989, 1991 by Jef Poskanzer.
Modified 26 November 1994 by Mike Burns, burns@chem.psu.edu

Table Of Contents

NAME
SYNOPSIS
DESCRIPTION
SEE ALSO
HISTORY
AUTHORS