NAME
pcompos - composite RADIANCE pictures.

SYNOPSIS
pcompos [ −h ][ −x xres ][ −y yres ][ −b r g b ][ −lh h ][ −la ] [ −t min1 ][ +t max1 ][ −l lab ][ =SS ] pic1
x1 y1 ..
or
pcompos [ −a ncols ][ −s spacing ][ −o x0 y0 ][ options ] pic1 pic2 ..

DESCRIPTION
Pcompos arranges and composites RADIANCE pictures and sends the result to the standard output. Each
input picture must be accompanied by an anchor point (unless the −a option is used, see below). This
anchor point is the usually position of the picture’s left lower corner in the final output, but can be changed
for individual pictures with an =SS option, where S is one of ‘−’, ‘+’ or ‘0’, indicating the minimum, maxi-
mum or center of the image, respectively. (For example, =+- would indicate the anchor is relative to the
right lower corner, and =-0 would indicate the anchor is relative to the center of the left edge.) Negative
anchor coordinates result in the input being cropped at the origin. By default, the size of the output picture
will be just large enough to encompass all the input files. By specifying a smaller dimension using the −x
and −y options, input files can be cropped at the upper boundary. Specifying a larger dimension produces a
border. The −b option specifies a background color to appear wherever input files do not cover. The default
value is black (0 0 0). The −h option may be used to reduce the information header size, which can
grow disproportionately after multiple runs of pcompos and/or pcomb(1).

If input files overlap, later pictures will overwrite earlier ones. By default, input files are copied uncondi-
tionally within the output boundaries. The −t option specifies a lower threshold intensity under which
input pixels will not be copied to the output. The +t option specifies an upper threshold. These options are
useful for cutting around irregular boundaries in the input.

The −l option can be used to specify a label for a specific picture, which will be given a height determined
by the −lh option (default 24 pixels) and placed in the upper left corner of the picture. This label is gener-
ated by the program psign(1). The −la option instructs pcompos to label each picture automatically by its
name. This is particularly useful in conjunction with the −a option for producing a catalog of images (see
example below). The −l option may still be used to override the default label for a picture.

The −a option can be used to automatically compute anchor points that place successive pictures next to
each other in ncols columns. The ordering will place the first picture in the lower left corner, the next just
to the right of it, and so on for ncols pictures. Then, the next row up repeats the pattern until all the input
pictures have been added to the output. If the pictures are of different size, pcompos will end up leaving
some background areas in the output picture. There will also be an unfinished row at the top if the number
of pictures is not evenly divided by ncols. The −s N option will cause each image to be separated by at
least N pixels. The −o x0 y0 option specifies a nonzero anchor point for the bottom left image. If a nega-
tive value is given for ncols, then the rows are reversed, so they start from the top rather than the bottom.
This gives a more natural ordering in most cases, but has the side-effect of moving any labels from the
upper-left of each placed subimage to the lower-left. Also, the −o option is interpreted differently, where
the y0 value is number of pixels from the top (rather than the bottom).

The standard input can be specified with a hyphen (‘-‘). A command that produces a RADIANCE picture
can be given in place of a file by preceeding it with an exclamation point (‘!’).

EXAMPLE
To put a copyright label at the bottom of a picture:

    psign Copyright 1987 | pcompos inp.hdr 0 0 +t .5 - 384 64 > out.hdr

To make a catalog of images separated by white 10-pixel borders and ordered four images per row starting
from the top-left:

    pcompos -a -4 -s 10 -b 1 1 1 dog*.hdr > alldogs.hdr
NOTES
Since there is a limit to the number of open files and processes, large collections of images must be created in stages. Even if the system limit on open files is large, *pcompos* places an artificial limit of 1024 on the number of open files and/or processes.

AUTHOR
Greg Ward

SEE ALSO
getinfo(1), pcomb(1), pfilt(1), psign(1), rpict(1)