**Table of contents**

Notes

Introduction

Misc

Menu

Modes of graphic files

Files and clipboard

Picture levels

Displaying

Color and pixel editing

General editing of the current level

Paintbrush ‑ and bugs in Paintbrush)

Swopping red and blue

Swopping light grey with dark grey

Putting a standard palette in

Halve size black wins

`This image's palette is nonstandard and may upset mode‑conversions etc'

Clipboard

Make a color image into a black‑on‑white line drawing

Make all lines one pixel wide

Undoing

Composition files

Slide shows

**Notes:** ***TC*** = only for truecolor images, ***CE*** = only in color‑editing mode, ***NC*** = only when not in color‑editing mode.

Menu calls are described in this help as the corresponding alt‑sequences.

**Introduction**

The mouse left button operates on the currently selected level according to which key you are holding down while clicking or dragging. PPP can read and write PCX and BMP andf .GIF and .JPG files and the clipboard. It can call Paintbrush.

PPP currently can handle 16 images called levels #0 to #15. Of these, #0 is the background. You can select which to be currently handling. PPP displays these 16 images superimposed in numerical order with #15 on top and the background on the bottom. For each of these images:‑

‑ You can **hide** it, i.e. tell PPP not to display it.

‑ You can choose a `transparent color' for it: when displayed that color shows as transparent so whatever is behind it shows through it. Any transparent color that the background has, or the whole background if it is hidden, shows as black.

‑ You can move and rotate and scale each image except the background by:‑

‑ ‑ dragging it with the mouse.

‑ ‑ (for the current image) by a dialog box called from the menu.

‑ Normally each level's priority is the same as its level number.

If you alter any of their priorities, the menu option `sort by priorities' will reorder the levels and the reset the priorities, to change which image prints on top of which.. The images only show within the background. If you try to work without a background, PPP creates a white background 640\*480 pixels.

If you load more than one image off the same file, PPP handles them independently: it does not update them to match each other.

**Menu**

**Files and clipboard**

**alt F N** give this level a new blank image

**alt F R** read from file

**alt F W** write to file

**alt F M** rename file

**alt F D** delete file.

**alt F P** how paths?

**alt F C** get contents list for clipboard

**alt F O** examine image on clipboard

**alt F Q** read image from .clipboard

**alt F B** write image to (clipboard

**alt F G** read composing info

**alt F I** write composing infon

**alt L S** save assembled picture to file

**alt L V** assemble current level & level 15, save to file or to current level

**alt F X** exit

**Modes of graphic files**

**alt M 2** change any mode to 2‑color

**alt M 1** change any mode to 16‑color

**alt M C** change any mode to 16‑color with pixel chequering (takes a long time)

**alt M 6** change any mode to 256‑color

**alt M D** change truecolor to 256‑color with dithering

**alt M G** change any mode to 256‑color greyshades

**alt M T** change any mode to truecolor

**alt M B** change truecolor to 2‑color by the bwlimit value

**alt M 4** this truecolor image, if saved, is to be saved as 16 or 24 bit?

**alt M E** show PCX file header

**alt M S** show palette

**alt M F** show byte usage frequency

**F9** ***TC*** While F9 is down, display the image black and white, being black where all the color values < *bwsplit*.

Also, if the image is truecolor and is converted to 2‑color mode, it will be converted using this same rule.

While **F9** is down, the **down left right up arrow keys** alter*bwsplit*)by ‑10 ‑1 +1 +10 respectively.

**ctrl‑F9** Display the old value and ask for a new value of *bwsplit*.

**T** show the current level's full title in a message box.

Graphic images occur in these modes:‑ (bpp = bits per pixel)

- Screen images and PCX files: bpp = 1, 4, 8, 16, 24.

‑ .BMP files, and Windows BITMAPs: bpp = 1, 4, 8, 24.

‑ 16 and 24 are truecolor, the rest have a palette of 2, 16, 256 members respectively.

I store bpp=1 and bpp=4 images internally as bpp=8: their modes are shown as 1:8 and 4:8.

I store bpp=16 images as bpp=24.

Some programs cannot read bpp=16 mode PCX files.

When converting truecolor images to bpp=8, PPP converts to an image with 216 palette members, being 6 brightnesses of each color.

‑ Windows (or at least Windows 3.1.1) seems to have an undocumented non‑standard bpp=16 BITMAP mode which is used when truecolor images are stored in the clipboard.

**Displaying**

**F10** redisplay the screen.

**F11** switch between displaying assembled images and displaying current image only.

***NC***: **home / end** scroll to top/bottom,

***NC***: **pageup / pagedown** scroll up/down a windowful,

***NC***: **insert / delete** scroll up/down a short distance;

***NC***: **ctrl‑insert / ctrl‑delete** scroll left/right a short distance.

**ctrl‑H** hide/show current level.

**alt L H** hide/show current level.

**T** show the current level's full title in a message box.

**ctrl‑alt‑space** swop the current level's main and backup transparent colors.

**B leftclick** choose as the current ,level's transparent color the color of the clicked pixel.

**V leftclick** The current level is set to have no transparent color.

**B rightclick** Choose as the current levels backup transparent color the color of the clicked pixel.

**V rightclick** The current level is set to have no backup transparent color.

**R leftdrag** Rotate the current level around its current pivot point.

**S leftdrag** Scale current level so the mouse stays the same scaled distance from the images's current pivot point, which stays still.

**R S leftdrag**) Both of the above: the same point of the image follows the mouse.

(else) (leftdrag) ***NC*** Drag the current level's image about.

Note: The B R and S mouse options don't work with the background.

**any\_digit leftclick** Display the picture scaled up that many times (0 means 10), with the clicked point at top left corner.

**ctrl‑any\_digit leftclick** Ditto but scaled that many times down.

**R rightclick** Set that level's current pivot point (it is originally at the images's origin).

**alt L E** mark edges where same color overlaps same color in another level; overwrites level 15

**P** While this key is pressed, the current image disappears (or appears if it was hidden before), to show which image is current.

**F4** While this key is pressed, if the window display is truecolor, I assume that it is a Poser bump map in .BUM mode and display the image mid grey as if illuminated from the mouse towards the center of the window (not towards the center of the image).

**General editing of the current level**

**alt E G** swop greys, 16‑color mode only

**alt E B** swop red and blue, 16‑color mode only

**alt E P** standardize palette, 16‑color mode only

**alt E V** replace colors by complementary colors

**alt E L** make line drawing (not truecolor)

**alt E H** halve size (black wins) (not 256‑color or truecolor modes)

**alt E X** make lines 1 pixel wide, 2‑color mode only

**alt E N** make all non‑background pixels black or white

**alt E M** trim off edges which are all that level's transparent color

**alt E K** where the current background contacts another color, spread the other color by 1 pixel.

**alt E B** make white all pixels which are brighter than bwlimit

**alt E T** tiling

**alt E K** change the color of all background pixels to the color of any adjacent non‑background pixel (truecolor only)

**alt E O** flip top/bottom

**alt E I** flip left/right

**alt E 9** rotate 90 deg to right

**alt E A** copy current level to level 15, trying to correct the `barrel distortion' which some digital camera images have.

**alt E U** copy current level to level 15, converting it to Poser .BUM file mode.

**alt E F** copy current level to level 15, converting it from Poser .BUM file. .i(Warning: this may not reconstruct the original exactly.)

**alt E D** Write left button's selected color on all pixels whose color is right button's selected color.

**ctrl‑E** enlarge current level by its current zoom scaling; color shade pixels so enlarged pixels don't show as pixellation.

**The picture levels**

**alt L V** assemble current level & level 15, save to file or to current level

**alt L L** list current filenames & other info for the levels; options to handle them

**alt L O** show scale & origin & angle of current level

**alt L A** get .dialog box to display & alter the current level's details

**alt L P** sort levels by priority

**ctrl‑K** remove level 15's image

**ctrl‑L** list the current filenames and other info for the levels;

**L arrow\_keys** attention moves up or down the list of levels. options to handle them

**ctrl rightclick**) ***NC***: Whatever image the mouse pointer is clicked on a part of (not transparent parts or hidden), becomes current image

**F5** if not on level 15, place level 15 same as this level and go to level 15; if on level 15, go back to the level that you came from, if that was by key F5.

**Color and pixel editing**

**F2 and F3 together** start or stop editing pixel colors by mouse click/drag.

***CE TC***: change the color of the current pixel:‑

In color editing mode only the current level displays, unless you pressed F11.

**insert/delete** blue up/down;

**home/end** green up/down;

**pageup/pagedown** red up/down; if with **alt** by 4, if with **ctrl** by 16, if with **shift** by 64, if with **tab** to max/min, else by 1.

**Q** displays its current color values.

**ctrl‑alt‑alphanumeric** set brush size to: `1' to `9' = 1 to 9, `0' = 10; `A' to `Z' = 11 to 36.

The current brush position is marked by an outlined square if it is large.

If that square is bigger than Windows's allowed cursor size, not all the square shows, sorry.

**F1 leftclick** make up a Paintbrush \*.PAL file whose last entry is the color of the pixel that the mouse is on. Paintbrush's menu option `Options, Get Color' reads .PAL files.

**F12 leftclick** Display the color brightness numbers of the clicked pixel. The Paintbrush user can then select this color by clicking the last palette entry.

**A leftclick** ***TC***: Replace this pixel and the pixel to its right by their average.

**Z leftclick** ***TC***: Replace this pixel and the pixel below it by their average.

**U leftclick** ***TC***: Replace a 2\*2 dithered pattern by its average color.

**leftarrow leftclick** ***TC***: (Blur Edge): Replace all pixels of top edge of current brush position by intermediate between its two end pixels.

**uparrow leftclick *TC***: Ditto but the left edge of the current brush position.

**rightarrow leftclick *TC***: (Sharpen Edge): Replace all pixels of the top edge of the current brush position by whichever of its two end pixels its color is nearest to.

**downarrow leftclick *TC***: Ditto but the left edge of the current brush position.

**esc leftclick** Graded color wash

**D leftdrag** In the brush area, write left button's selected color on any pixel whose color is right button's selected color.

**ctrl rightclick  *CE***: choose a `current pixel'.

For any mouse button, ***CE***:‑

**F2 click** Set the clicked button's color to the color of the clicked pixel.

**F3 click** ditto, and call a Color Change dialog box to change the resulting color.

**E click** Call a Color Change dialog box to change the clicked button's color.

**F click** Flood fill: replace the color where clicked with this button's color as far as it spreads. **Esc** key halts the fill.

*else*, **CE**: set clicked pixel to clicked button's color.

*(In color editing mode in `show all levels' mode, pixels set by this means to the current level's transparent mode show that color and not what is behind them until that part of the screen is redrawn again The middle button works only if your mouse and its driver are set up in 3‑button mode.)*

**F shift click** Flood fill: replace any other color than the color where clicked with this button's color until this buttons color is reached. **Esc** key halts the fill.

*else*, **CE**: set clicked pixel to clicked button's color.

**Misc**

**alt E U** Undoing

**ctrl‑U** copy current level's image to the `Undo' buffer

**alt A** call Paintbrush

**alt H** help

**ctrl‑Z** assemble the images onto image 15.

**shift arrow\_keys** the current image moves.

**Options while list of levels show**

The current level is shown in red.

Clicking the left mouse button on a level's entry line selects that level.

Dragging the right mouse button changes the order of the levels (but not the background).

**D** get dialog box to alter settings of current level (not background)

**F** key switches the filenames between `display full pathname' and `display local name only'

**H** key hides or reveals the current level

**Dialog box to display and alter that level's settings**

This dialog box changes the current level's: offset & scale & angle relative to the background, and its transparent color (listed as a hex value). It also displays and alters the general display offset and zoom scaling. Attempts to set illegal values are ignored or corrected.

**Paintbrush**

‑ PPP has a menu option to call Windows Paintbrush on the file that you are currently working on. When you have finished with this Paintbrush call, exit it; when using PPP it is easy to let multiple leftover calls of Paintbrush accumulate in your Windows.

‑ Paintbrush seems to produce PCX files with color mode the same as your Windows is working at. When this is truecolor, I have come across these bugs:

‑ When writing a PCX file, using bpp=15 mode (5 bits per color), but calling it bpp=16, causing errors in the color intensities if you read it as genuine bpp=16. For this reason PPP reads and writes bpp=16 PCX files as bpp=15.

‑ When writing to the clipboard, omitting the last pixel of each bufferful, causing white dots on the image. *(Windows 95 Paint alias Paintbrush has corrected these bugs.)* *(It reads bpp=24 but not bpp=16.)*

**Swopping red and blue**

Paintbrush mode 5 for 16‑color has red and blue the other way round from other modes in the order of its color planes. That sometimes gets red and blue swopped, e.g. when Paintshop Pro operates on a mixed batch of PCX files. This menu option swops them back.

**Swopping light grey with dark grey**

Whoever in ZSOFT wrote Paintbrush got the greys swopped,so 16‑color pictures made by Paintbrush are sometimes displayed by other graphics packages with light grey instead of dark grey and vice versa. (The correct image pixel assignation is 7 = light grey, 8 = dark grey.) This menu option swops them back.

**Putting a standard palette in**

On one of my PC's Paintbrush had a habit of writing the palette zeroed when saving to a .PCX file, so the image when read back appears as black.*(Warning: if this happens, do not call Save in Paintbrush on the blacked image, or Paintbrush will standardize the image in its own way, by changing all the image pixels to 0 [= standard palette black].* Also, sometimes 2‑color mode files are displayed as black and white always and sometimes as whatever the palette is. This menu option puts a standard palette in: it does not change the image pixels, so some of the colors of the image may change.

**Halve size black wins**

Halve the size of a 2‑color image, When pixels are amalgamated, if any of the component pixels are black, the result is black rather than taking the average, so thin black lines on white background are not lost.

**This image's palette is nonstandard and may upset mode‑conversions etc.**

If you get this warning on reading a graphic file: To put a standard palette in and also change the image pixels to match, call the menu option `Modeconv', `any to 16‑color'.

**clipboard**

PPP can read and write the clipboard. (Warning): Windows 3.1.1 and perhaps other versions store images on the clipboard with color accuracy reduced to the color accuracy of the screen mode that your Windows is currently using, but compressing truecolor images

to a nonstandard 16‑bit or 15‑bit mode.

**Make a black‑on‑white line drawing**,

This option makes a color image (not truecolor) into a black on white line drawing. Unlike CorelDraw's version, it can cope with the common .Paintbrush feature of making a halftone by chequering the pixels alternately in two colors.

**Make all lines one pixel wide**

This option, if called on an image which is black lines on white, thins all lines down to one pixel wide, curing the irregular line width and battlement effect often seen in drawings scanned in by a scanner. The deleted pixels are not blanked but turned to red, so you can see what happened and revise the result with Paintbrush.

**Undoing**

Swop the current level image with the `Undo' backup image, if the two have the same size and type. The `Undo' backup image is currently set by:‑

Flood‑fill (mouse click with F key)

**ctrl‑U** copies the current level's image to the `Undo' buffer

**Save assembled picture to file**

This option copies the assembled picture onto a file, with the background image overlaid with copies of the other images as you see on the screen.

**Composition files**

Files with suffix `.PPP' (Ppp Picture comPose) contain a list of names of files to load a set of image levels from, and their offsets and relative scales and angles and transparent colors. They let you save a composition of images between editing sessions.

**Graded color wash**

This can be called on any level except 15. It assumes that the current level and level 15 have the same current origins and scales and rotation display angles. (If they are not, press **F5** twice.) It paints a graded color wash onto an area of the current level, according to this information as stored in level 15:‑ Area painted on = area corresponding to that part of level 15 which is **not** in level 15's current backup transparent color Datum points = corresponding pixels in level 15 which are not in level 15's transparent color and not in level 15's backup transparent color. In each of the 3 color intensities (blue, green, red) it fits a polynomial to the datum points. If the process takes too long, pressing .**Esc** will quit out of it.

**Slide shows**

If a file with suffix `. sli' contains a list of graphics file names, one per line, PPP can read this file and display the files named as a slide show.The pop‑up menu "Slides" has these options:‑

"file to read slide file names from" start reading a slide file, to display the slides in the current level.

"file to list slide files names on" open a file to list slide file names on.

"stop listing slide file names" if such a file exists, close it.

Thereare these related key presses:‑

**right-arrow key** display the next slide.

**next but one key to right of L** (apostrophe on UK keyboards) go back to the first slide.

**next but one key to right of M** (usually fullstop) list the current level's current file's name on the current "output a slide list" file.

**next but two key to right of M** (usually slash) ditto and ask for a caption for that slide.

A slide file can be read by dragging it from Windows Explorer or "My Computer" into a PPP window, or by binding the filename suffix `. sli' to the file PPP.EXE .

Format of . sli file:‑

A . sli file is in text format and can be edited with a text editor.

Each line should be a graphics file name, with no spaces or tabulates added at the beginning or end.

If a **#** is appended to a filename line, the next line is treated as a caption which is displayed below the slide.

If a **#** is appended to a caption line, the next line is treated as another line of the caption.

There must be no more than 4 lines of text in a caption.

If a filename is a directory name, that directory is used to read relative‑type filenames from then on in that . sli file.