Palmcrust®

Just Jigsaw

For Palm®, Sony® CLIÈ$^{\text{TM}}$ and compatibles

User manual

Version 1.1.1
Michael Glickman
palmcrust@yahoo.com
August 2004

Content

Introduction

Should I really explain what is a jigsaw puzzle ? For the current implementation the objective is the following: "All frames on the desktop are parts of a single picture. You need to reconstruct the picture by placing each frame into its proper location on the board".

The application allows processing pictures larger than screen size. You can have a virtual screen of up to 240x240 on a regular Palm, or 480x480 on Sony CLIE™ , so that each time the actual screen is mapped onto a particular virtual screen area.

The application provides a variety of game options and preferences. It comes with demo images and allows using your own pictures in PalmOS™ bitmap or PictureGear™ formats . If your handheld has an expansion slot, the application will take advantage of that by reading image directly from an external memory source.

The latest release of the product, additional images and more information are available from the application site http://justjigsaw.tripod.com

If you have questions, comments, or problems, please don't hesitate to contact me: palmcrust@yahoo.com
Please place 'jjig' somewhere in the subject for easy filtering.

Terms

Actual screen size: screen size provided by the hardware, excluding top (frame area): 160x145 on a regular Palm, or 320x280 on Sony CLIE™.

Buttons: 6 buttons located on the bottom of your Palm, typically used for *Date Book*, *Address Book*, *Page Up*, *Page Down*, *To Do List* and *Memo Pad* by the Launcher.

Launcher: Top PalmOS™ application that shows application icons by categories and starts other applications.

Jog Wheel: a wheel located on the left side of Sony CLIE ™ handheld used for scrolling lists, text etc.

Main form: the form used form playing as opposed to additional forms (Preferences, Game Options etc)

Scrolling: changing view spot on the virtual screen.

Selection mode: STOP or GO. Selection type depends on where you tap the frame, and can be changed by entering a button combination or tapping a Top Control. It affects stylus, jog wheel and some button operations. See Operating frames. for more information about selection mode.

Steering: moving or rotating a frame.

Target: a board place holder where current frame is expected to be successfully placed.

Top Controls: Controls ('soft buttons') located on the top of the main form.

Virtual File System (VFS): PalmOS™ extension that allows reading from an external memory card.

Virtual screen size: screen size of the play area (desktop). If virtual screen size exceeds the actual size, you can specify the **view spot:** which part of the virtual screen is shown on the actual screen

Operating frames

You select a frame usually by tapping it. A frame can be selected for **STOP** (**red** borders), or **GO** (**green** borders), referred below as *selection mode*. On a monochrome screen, the selection mode can be identified from the appearence of `Move`, `Rotate` and `Stop` controls (see Top Controls).

In **STOP** mode you can rotate frame, or change selection, while in **GO** mode, a frame can slide, or jump. Use stylus, buttons, or Graffiti® for steering.

When a frame gets reasonably close to its board location, it is *placed* on he board. Once the frame is placed, you can't move it any longer.

Using Stylus

Stylus (pen) provides the most intuitive way to operate frames.
To select frame in **GO** mode, tap a frame near its centre, for **STOP** mode, tap it near the border.

Short move (slide).
Tap a frame near its centre, so that its borders are **green**. Drag frame in the appropriate direction.

Long move (jump).
Tap a frame near its centre, so that its borders are **green**. Lift pen and tap anywhere on the screen (you might need scrolling it). The frame moves so that its centre is positioned in specified place.

Rotation.
Tap a frame near its borders, so that the borders are **red**. Drag in the direction of rotation.
For `Right Angles` rotation type (see Game Options), you can only rotate 90 degrees in each direction.

Changing current frame.
When frame borders are **red** , just tap another frame. .

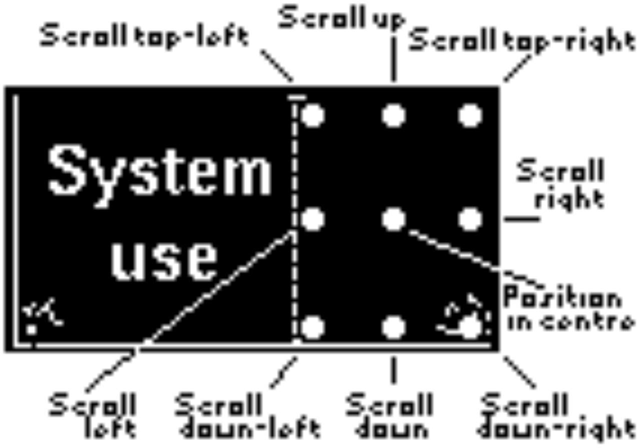
Hint: To avoid hitch hiking (a frame is dragged along the board without a particular target, until is is placed somewhere), placing a dragged frame is harder than placing a 'still' one. If you haves trouble with placement, use buttons for moving in the proximity of the target.

Using Graffiti®

Graffiti® can be enabled for *scrolling*, or *steering* in Preferences form . In both cases only numeric area is used, therefore you can still use Graffiti for menu shortcuts, and have a smaller chance of tapping `Launcher` button accidentally. For your convenience, right silk−screen buttons (`Calc` and `Search`) are disabled, if Graffiti is enabled for the application.

Scrolling.

If virtual screen size is bigger than the actual one, you can use Graffiti for moving within the virtual screen. For scrolling, just tap numeric area as shown on the following diagram:

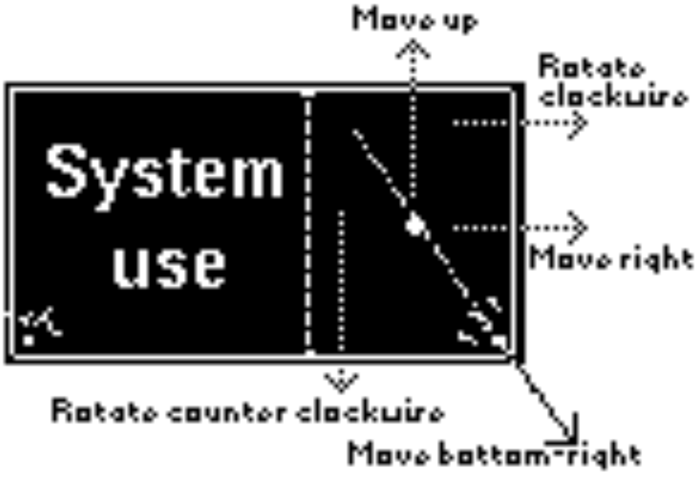


When Graffiti is enabled, it always operates for steering, when virtual screen is same as the actual one.

Steering.

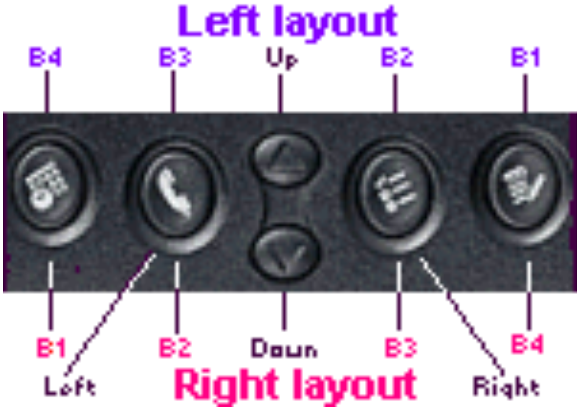
An advantage of steering with Graffiti is that you shouldn't care about the selection mode: the frame moves or rotates, depending on how you start moving the stylus: if you start from the centre of numeric area, or moving toward the centre, frame moves, otherwise it rotates. Once the mode has been defined, it doesn't change until you lift the pen (you can even move it off Graffiti area), however you have to start from the numeric area.

If the above explanations don't appear quite clear, the following sketch might be helpful:



Operating Buttons

You can choose Left or Right button layout in Preferences form , as shown on the following diagram:



Note that while button numbers are affected by the layout, names **`Left,`** **`Up`**, **`Down`** , and **`Right`** always refer to particular button locations.

While entering a button combination, first button should be held, while pressing the following one, e.g. for **`B1+Up`** , press **`B1`** , and while holding it press **`Up.`**

Main form

The following actions depend on selection mode:

GO selection mode:

`Up`	Move current frame 1 pixel up.
`Down`	Move current frame 1 pixel down
`Left`	Move current frame 1 pixel left.
`Right`	Move current frame 1 pixel right
`Left+Up`	Move current frame up−left (1 pixel in both dir−s)
etc.	

STOPselection mode:

`Up`	Turn 1.4 degrees clockwise.
`Down`	Turn 1.4 degrees counter clockwise.
`B2+Up`	Turn 11.25 degrees clockwise
`B2+Down`	Turn 11.25 degrees counter clockwise.
`B3+Up`	Turn 90 degrees clockwise
`B3+Down`	Turn 90 degrees counter clockwise

For rotation type `Right Angles (see` Game options `),` angles are 90 degrees for each combination.

Both modes:

`B1+B2`	Flip
`B1+Up`	Undo
`B1+Down`	Select next frame
`B1+B3`	Toggle selection mode
`B1+B4`	Preview (while pressed)
`B4+Up`	Scroll view up
`B4+Down`	Scroll view down
`B4+Left`	Scroll view left
`B4+Right`	Scroll view right

Load Image form

`B1`	Game Options
`B2`	Image Source
`Up`	Select previous list item
`Down`	Select next list item
`B3`	Cancel
`B4`	Select image

Help

`Up`	Scroll help text 1 line up
`Down`	Scroll help text 1 line down
`B1+Up`	Move 1 page up
`B1+Down`	Move 1 page down
`Left`	Go back (previous item)
`Right`	Go forward (next item)
`B4`	Leave help − when pressed from the help index; Go to help index − when pressed elsewhere.

Other form

`B3`	Cancel

B4	Accept

Operating Jog Wheel

The following assumes that both jog wheel and back button are enabled (see CLIE^{TM} Special Form).

Main form

The following actions depend on selection mode:

GO selection mode:

Up	Move current frame 1 pixel up.
Down	Move current frame 1 pixel down
Up pressed	Move current frame 1 pixel left.
Down pressed	Move current frame 1 pixel right

STOPselection mode:

Up	Turn 1.4 degrees clockwise.
Down	Turn 1.4 degrees counter clockwise
Up pressed	Turn 11.25 degrees clockwise
Down pressed	Turn 11.25 degrees counter clockwise

Both modes:

Back button	Toggle selection mode

Load Image form

Up	Select previous list item
Down	Select next list item
Press	Select image
Back button	Cancel

Help

Up	Scroll help text 1 line up
Down	Scroll help text 1 line down
Up pressed	Move 1 page up
Down pressed	Move 1 page down
Back button	Leave help − when pressed from the help index; Go to help index − when pressed elsewhere.

Other forms

Press	Accept
Back button	Cancel

Additional forms

Load Image Form

This form allows selecting to start another puzzle with another picture and/or game options. The form displays the list of images. Images located on an external source are painted blue.

The buttons below are:

⊙	Game Options
✎	Image Source
✕	Cancel
✓	Accept

You can specify additional directories and/or change 16bpp processing mode by tapping *Image Source* button, so that while returning, the list may change. The bottom line shows size and depth of the selected image. You can either tap *Accept* button to continue without changing game options, or press *Game Options* button to proceed with selected image and modified game options. You can't change game options within a game.

Preferences Form

Select `View / Preferences` to show the form. Setting are activated straight away. Default values are given in brackets.

Enable Sound (true)
Enable game sound. If game sound is enabled, the volume is determined by Prefs **`Game Sound`** parameter (`Prefs` is a PalmOS^{TM} standard accessory). If `Prefs Game` Sound is *off,* but the sound is enabled for the correct application, low volume is used.

Show content while dragging (false)
Show frame content (not border only) when moving or turning. Enabling this option may result in a big drawing latency.

Store completed puzzle (false)
If you quit when puzzle is completed, it won't be saved, so that next time the application starts with the image list. If you enable this option, a completed puzzle is saved, and appears, when application restarts.

Auto−scroll (true).
Shown, if virtual screen exceeds actual one. When enabled, virtual screen view spot is shifted automatically, when current frame goes out of sight.

Numeric Graffiti area use (Scroll)
One of two ways of using Graffiti area (see Using Graffiti), or *Disabled* − Graffiti strokes are processed

by the system.

Bottom Button Layout (Right)
Button layout (see Operating Buttons), or *Disabled* − buttons are processed by the system.

Image Source Form

This form is available if the handheld supports 16bpp and / or Virtual File System (VFS) for reading from an external memory.

Use 16bpp image if available.
This option is enabled by default, however you might turn it off for using a 256−colour image, even when a true−colour equivalent is available.

Enable Virtual File System.
Disable VFS only if you have troubles with your expansion slot.

VFS Image file directories.
In case VFS is enabled, you can set up to 3 directories to look up the images. The directories are processed in specified order.

Game Options Form

You can set game options by tapping `Game Options` button in Load Image form.
After game started, you can view options by selecting `View/Game Options` menu item.

Virtual screen

Allows selecting virtual screen size. It can't be less than the maximum of image size and actual screen size. The maximum values are:

Regular Palm 16bpp:	180x180
Regular Palm 8bpp:	240x240
Sony CLIE (any):	480x480

Use **Auto−size** to let the application choose a convenient screen size for selected image.

Low resolution where possible. (Sony CLIE only)
When enabled, high screen resolution is used only for big virtual screen sizes, where low resolution cannot be used. If disabled, high resolution is always used.

Cut strips

Number of vertical and horizontal strips, the image has to be cut into. Limits are selected according to current image size.

Rules

Time limit (in minutes).
For an unregistered user time limit is always 10 minutes, so that this option can be maintained by a registered user only.

Flip.
if enabled, some frames appear flipped. To place them on board, you need to 'unflip' using *Flip control* (see Top Controls), or *Flip button combination* (see Operating Buttons).

Hide cutting lines
When ticked, the board appears 'plane' without cutting lines.

Straight cuts
Avoid curly cuts along specified direction(−s). With straight cuts along both directions, frames are rectangular, which makes it really hard. *Straight cuts are applied to small frames even though this option might not be set.*

Rotation.

None	disables rotation originally and in the game;
Right angles	originally each frame is turned on a multiple of 90 degrees (i.e. 0, 90, 180 or 270 degrees); only 90 degree turns are allowed within the game
Free	the default mode: rotate at any angle at any time

CLIE™ Special Form

This small form allows to enable, or disable using Jog Wheel with the application.

If Jog Wheel is enabled, you can also specify, if back button is also used for the application. If back button is enabled, you have to use *Launcher* silk screen button (top left near Graffiti) to quit the application.

See Operating Jog Wheel   for using jog wheel and back buttons with application.

Status Form

Displays an information related to the environment and application status:

PalmOS	Version of the operating system
Hi Res	Version of CLIE^{TM} HR library or U/A
VFS	Version of VFS library or U/A
Jog wheel	Typ1 (no Back Button), Typ2 (has Back Button) or U/A
Moves	If within a game, shows number of moves from start of the game.
Screen res. depth	Size of digitizer screen area, and maximum depth as `width x height x depth`
Memory:	Total memory, free memory and per cent of free memory: Dynamic: dynamic heap (short time memory) Storage: storage heap (long−time memory, data bases) ROM: read−only memory (typically PalmOS^{TM} standard software)
Bitmap:	Image size, depth and location: First line: width x height x depth palette_type where palette_type (for depth 8 only) is std. colours − PalmOS^{TM} standard (web−safe) palette colour table − custom colour table Second line: database(BMP) or image(PictureGear^{TM}) name Third line: location *Local storage* − memory card 0 *Memory card 1* − memory card 1 *Ext memory* − external card with empty volume number <vol. number> − volume number if non−empty
Virtual	Virtual screen size (width x height).

Miscellaneous

Puzzle Menu Items

Puzzle / New Puzzle − brings up Load Image Form that allows selecting a new image and/or game options. Current game is lost.

Puzzle / Restart − restores current puzzle to its original position.

Puzzle / Terminate − give up. All frames are placed and puzzle terminates.

Pausing

There is no special *Pause* command: the game pauses (game timer stops) at any time when you leave the main form (e.g for preferences, status, or game options), or switch off your handheld. Note that image view doesn't pause the game timer.

Top Controls

The controls on the top of the main form are used as the following:

▣ − scroll
✥ − move

⇤ − undo
⟳ − rotate
⛔ − stop

↔ − flip
▨ − view

Scroll.
If virtual screen is bigger than the actual one, you can change the view spot by tapping scroll control in appropriate place, e.g. tapping upper left corner of the control shows the upper left part of the virtual screen.

Undo.
As the name suggests. Up to 20 consecutive moves can be withdrawn.

Flip.
Vertical flip. Appears if flips are enabled (see Game options).
Hint: horz.flip = vert.flip + 180 degrees turn

Move.
Tap control in the centre for changing selection mode to **GO**. Tapping outside the centre, in addition, moves current frame in appropriate direction. The speed increases while approaching the control boundary.

Rotate.
Tap control in the centre for changing selection mode to **STOP**. Tapping the bottom part rotates frame

clockwise, while tapping the top part rotates counter clockwise. Rotation speed increases from right to left.

Stop.
Replaces *Rotate* control, in case rotations are disabled (see Game Options). Used for changing selection mode to **STOP.**

Apart from their main purpose, Move, Rotate *. and* Stop controls can be used as an indication of current selection mode on a monochrome screen: while in **STOP** *mode,* Rotate *or* Stop *control is bright, and* Move *is dim, whereas in* **GO** *mode, controls look in the opposite way.*

View.
Tap the control to view the picture. An image exceeding screen size is shown scaled 1 : 1.5 (i.e. squeezed 1.5 times) . Lift the stylus for returning to game.

Registration

Do you like the application?
Do you wish a new version to appear soon ?
Do you expect another software from the same developer ?

Please, register only if you answer yes on any of the above questions.

As a matter of fact, a registered user has the following advantages:

- no forced time limit.
- access to custom PalmOS Bitmap® and PictureGear® Pocket images.

Once you purchased the license, it lasts forever for the given release on your handheld. If you have to replace your unit, no panic: you can get another code twice within a year after the registration for same user name. You will be able to use the registration code for all past and future releases of the application.

Usually you can purchase the product from the site you download it.

See also:

http://www.palmgear.com
http://www.handango.com

http://www.pdassi.de

Using Custom Images

FYI. You must be a registered user to use custom images with the application.

Formats of custom images and size limitations

The application can read a custom image form a local or external memory card.. Two types of image formats are processed:

- PalmOS ™ bitmap images: this resource data base consists of images in compressed or uncompressed PalmOS bitmap format It also includes string resources that give readable names to bitmaps. Several images are can be stored in a single resource file.
- PictureGear ™ Pocket Format (PGPF) images, created and maintained by Sony CLIE ™ bundled software. A PGPF resource presumably stores a single image, however it may have several bitmaps for different size and depth. A PGPF image can be processed on a regular Palm, provided its size satisfies PalmOS limitations (see below).

Size limitations:

Sony CLIE: 480x480 (any depth)
Images with size exceeding 64K must be stored in PGPF format.
Using big true colour images is not recommended because of possible dynamic heap overflow (see Known Bugs and Troubleshooting).

All others: 180x180 for true colour (16bpp) images, and 240 x 240 for 256−colour (8bpp) images or 16 colour (4bpp) imagge
A PGPF image can be used if it satifsies size limitations for its depth.

On a monochrome screen, PGPF images are converted to 4bpp greay scale. In all other cases an image can be processed only if its depth is supported by the current device.

Creating custom images

Since PGPF images can be used straight away, this section concentrates exclusively on PalmOS bitmaps.

What you need

If you are novice to Palm Computing Platform, just get 'Just Jigsaw Custom Image Toolkit' from Jjig site Currently this is just a collection of 'third party' software (with some bug fixes) with some sample files and hints, however I am working on a GUI interface (for both Linux and Windows) that will make it convenient for someone who can't think of anything else.

Alternatively, you can download the software from their official cites:

Palm resource compiler (pilrc) (Wes Cherry & Aaron Ardiri)**:** http://www.ardiri.com
Palm database archiver (par) (David Williams): http://djw.org

If you are a PalmOS ™ developer, you might prefer using *build−prc* (a *prc−tools* component) to *par,* simply because this doesn't require any additional downloading. The examples with *build−prc* are also given below.

Now do it !

You can use Linux, cygwin, MS Windows™ or even MS DOS.

Images

You obviously need images. The accepted formats are .bmp (MS Windows™ Bitmap) and .ppm (Portable Pixmap).
In case of using .bmp files the recommended format is 8bit indexed .bmp (for 8bpp PalmOS bitmap) or RGB .bmp (for 16bpp PalmOS bitmap).
Don't use RLE encoding with .bmp , and don't rely on pilrc depth reduction: use your graphic editor for that.

Create a separate directory (*folder* in MS jargon) and place all required images there.
This is where you should also place a source file for pilrc, discussed below.

Pilrc source file

Use any text editor (like Notepad) to create a file with extension .rcp (e.g. `myimages.rcp`) that contains something like this:

```
VERSION ID  1 "1.0"

STRING 1 "Rose"
BITMAPCOLOUR ID 1001 "rose256.bmp" COMPRESS COLOURTABLE
BITMAPCOLOUR16K ID 2001 "rose.bmp" COMPRESS

STRING 2 "Phoenix"
BITMAPCOLOUR    ID 1002 "phoenix_grey.bmp" COMPRESS
BITMAPGREY16    ID 2002 "phoenix256.bmp" COMPRESS

STRING 3 "Reptiles (M.C. Escher)"
```

```
    BITMAPGREY16    ID 1003 "reptiles_grey.bmp" COMPRESS
```

The top line defines version, currently 1.0.

The following lines define the images. As you can guess, this resource file will have three images named "Rose" "Phoenix" and "Reptiles"

Image "Rose" has two bitmaps: one 8bpp (BITMAPCOLOUR), another 16bpp (BITMAPCOLOUR16K). Both are compressed to save space. Keyword COLOURTABLE specifies that a 256−colour image for "Rose" uses a custom colour table (*read Using Custom Palette section before use!*) . Since 4bpp bitmap is not provided, this image can't be processed and will be ignored on a monochrome screen.

Image "Phoenix" provies a 4bpp bitmap (ID 1002) to use with monochrome devices. It also has a separate 8bpp bitmap for colour devices. The 8bpp bitmap was created using a www−safe palette, therefore there is no need for a custom colour table. Www−safe palette is supported by some graphic editors like *gimp* in Linux This a safer way of using 8bpp images, though it may not reproduce colours correctly.

Finally "Reptiles" has a single 4bpp bitmap that will be used with all supported devices.

As you can see from the example, there is no need to supply images for all depths Usually you can use a 4bpp for all devices supported by the application and a 8bpp for all colour devices. However a 8bpp image is not processed on monochrome devices, while a 16bpp can be processed only on a handheld that supports true colours.

Notice the resource IDs! The string resources are numbered from 1 forward (1, 2, 3 etc). This is a preferred way of choosing name IDs. *You must use consecutive numbers starting from 1 if you want to read your images from an external source.* The bitmap IDs have to be **always** 1000+nameID, 2000+nameID , etc. Thus, in the above example, 1001 and 2001 are bitmap IDs for "Rose" (image 1), while 1002 and 2002 are bitmap IDs for Phoenix (image 2).

Getting all done

Once you created the .rcp file, you need to compile it and bind into prc.

Place `pilrc` and `par` into one of PATH directories (usually /usr/local/bin in Linux, /usr/bin in cygwin, C:\WINDOWS\COMMAND or C:\WINNT\COMMAND in MS Windows, and C:\DOS in MS DOS).

Go to shell / command prompt and change to the directory containing the images and rcp file.
Now you say:

Linux or cygwin:

```
   pilrc −q  myimages.rcp
   par   c −a  "resource"  myimages.prc  "My Custom Images"   IMGE  JjIm *.bin
   rm   −f *.bin.
```

MS Windows or MS DOS:

```
   pilrc −q  myimages.rcp
   par   c −a  "resource"  myimages.prc  "My Custom Images"   IMGE  JjIm *.bin
   del  *.bin.
```

Comments.
First step compiles file `myimages.rcp` in the previous step and produces several .bin files.

The second step creates a Palm resource file `myimages.prc` from the bin files obtained from the compilation.
The file is created with data base name is "My Custom Images", type 'IMGE', creator ID 'JjIm' (for Jjig Images).

- **Try using a reasonably unique data base name to avoid clash.**
- **You must use 'IMGE' for data base type and 'JjIm' for creator ID !**

The last step removes all bin files.

After that you need to upload the .prc file to your Palm (with pilot−xfer or Palm Desktop). If the file is OK, you new images wll be listed when you restart Jjig .

−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−

Build−prc users need to replace the second line in the above examples with

```
   build−prc −t IMGE myimages.prc "My Custom Images" JjIm *.bin
```

Actually pilrc has −ro option to create a prc image straight away, though it doesn't appear to work. However −ro provides a neat way to create the images with build−prc:

```
   pilrc −q −ro myimages.rcp  myimages.ro
   build−prc −t IMGE myimages.prc "My Custom Images" JjIm myimages.ro
   rm myimages.ro
```

And even simpler.

If you are tired of typing the above command lines, the following scripts can reduce .prc creation to a single line.

Linux or cygwin.

The following script assumes using ash − compatible shell (ash, bsh, bash).
For csh and tcsh some minor changes are required.

Store the shell script below as `/usr/local/bin/jjigCreateImage.` The file should have attributes 755:

```
   chmod 755 /usr/local/bin/JjigCreateImage
```

Now, to create Palm resource (.prc) file you change to the appropiate directory and type a single line like this:

```
   JjigCreateImage    myimages  "My Custom  Images"
```

This call compiles file `myimages.rcp` and creates a resource file `myimages.prc` with data base name

"My Custom Images".
Note that you don't specify a file extension in the command.

The text of jjigCreateImage:

```
! /bin/bash
#---------------------------------------------------------------------
# Create an image file (bash version)
# Synopsis:
#       JjigCreateImage prc_name db_name

filename=""
dbname=""

if (test $# = 1) ; then
    filename="$1"
    dbname="$1"
elif (test $# = 2) ; then
    filename="$1"
    dbname="$2"
fi

if ((test "$filename" = "") || (test "$dbname" = "")); then
    echo ""
    echo "Usage"
    echo "  JjigCreateImage prc_name db_name"
    echo ""
    echo "  where:"
    echo "  prc_name − name of resource file without an extension"
    echo "  tdb_name  − name of the data base"
    echo ""
    echo "Example"
    echo "  JjigCreateImage myimages \"My Custom Images\""
    echo ""
else
    echo "Compiling resource"
    pilrc −q $filename.rcp
    whichpar=`which par 2>/dev/null`
    if (test "$whichpar" = ""); then
      echo "Calling build−prc"
      build−prc −t IMGE $filename.prc "$dbname" JjIm *.bin
    else
      echo "Calling par"
      par c −a "resource" $filename.prc "$dbname" IMGE JjIm *.bin
    fi
    rm −f *.bin
fi
```

<u>MS Windows or MS DOS.</u>

Store the script below as jjimage.bat in one of PATH directories (e.g. C:\WINDOWS\COMMAND, C:\WINNT\COMMAND C:\DOS) .

Now, to create Palm resource (.prc) file you change to the appropiate directory and type a single line like this:

jjimage myimages "My Custom Images"

This call compiles file `myimages.rcp` and creates a resource file `myimages.prc` with data base name "My Custom Images".
Note that you don't specify a file extension in the command.

The text of jjigimage:

```
REM ---------------------------------------------------------------------


@ECHO OFF
REM ---------------------------------------------------------------------
REM Create an image file (MS DOS version)
REM Synopsis:
REM jjimage prc_name db_name
REM ---------------------------------------------------------------------

SET FILENAME=%1
SET DBNAME=%2

if "%FILENAME%" == "" goto BadArg

if "%DBNAME%" == "" goto DBNameImplicit
goto DBNameExplicit

:DBNameImplicit
SET DBNAME=%FILENAME%

:DBNameExplicit

@ECHO Compiling resource
pilrc –q %FILENAME%.rcp

@ECHO ------------------
@ECHO Calling par
par c –a "resource" %FILENAME%.prc %DBNAME% IMGE JjIm *.bin
DEL *.bin
goto Exit
:BadArg
@ECHO .
@ECHO Usage
@ECHO jjimage prc_name db_name
@ECHO .
@ECHO where:
```

```
@ECHO prc_name − name of resource file without an extension
@ECHO tdb_name − name of the data base
@ECHO .
@ECHO Example
@ECHO jjimage myimages "My Custom Images"
REM −−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−

Exit:

REM −−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−−
```

Using custom palette (colour table)

Custom palette is used with indexed (8bpp) images. It means that the bitmap stores RGB for each colour index, instead of using standard RGB values for indices.

As it was mentioned, using custom palette is generally not recommended, because:

- According to Palm OS documentation, it can significantly affect the performance.
- With custom colour table, some application icons might appear with wrong colours.

However, you might wish to use it, if your image looks really weird with standard (www−safe) palette. All demo PGPF files have custom palette, and though it was not my original intention, I had to include the support custom colour table, so that it is now available for both types of images.

Be warned that official pilrc release has a bug that prevents from creating a compressed image with custom colour table, therefore your choice will be one of the following:

- Use uncompressed images
- Get pilrc fixed from Just Jigsaw site

Your bitmap needs to use no more than 240 colours (with 234 it will be event better). This is because the remaining 16 colours are reserved to use by the application.

To include a custom colour table, you just code COLOURTABLE with BITMAPCOLOUR directive as shown in the example above.

Known Bugs and Troubleshooting

Here are some known bugs and other problems you might come across. This list is probably far from being complete. Expecting your feedback.

Bitmap creation error

You may get "**Bitmap creating error 502** " (please contact me with another code!) This means that you don't't have enough memory with creating bitmaps, or the memory is fragmented. To fix the fragmentation problem, just get out to Launcher and restart. If it still doesn't help, consider reducing the depth from 16 to 8bpp, or reducing bitmap size. Be aware that this application requires much more dynamic heap space than PictureGear™ Pocket does.

Distorted screen or redundant screen redrawing.

Well, edundant screen redrawing might get out of my control one day. It could be a bit better, and I hate any patches, due to unpredicted PalmOS behaviour (what I observe on the Emulator, is not what I get on my Palm).

As to distorted screen, it most probably has been fixed, it case it hasn't, use View/ Redraw Screeen .

Bad colours with custom palette.

I must admit that custom palette support is still in beta. ANyway, I am not going to redesign the icons, simply because I don't believe it is so crucial.

Looks bad on a monochrome screen, especially with Handera 330

Increase brightness, if possible. If it doesn't help ... get a colour device, or at least Sony CLIE 415. :=)
As to Handera 330 1.5 high resolution, it definitely needs special care...