

JIV_e

Jürgen Weinelt

COLLABORATORS

	<i>TITLE :</i> JIV_e		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY	Jürgen Weinelt	July 1, 2022	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	JIV_e	1
1.1	JIV Documentation	1
1.2	Author And Copyright	1
1.3	Introduction	2
1.4	Supported Image File Formats	3
1.5	Supported Display Hardware	4
1.6	Distribution	4
1.7	Shareware License	6
1.8	News	7
1.9	Installation	7
1.10	The JPEG CODEC	8
1.11	SHELL Usage	8
1.12	Program Parameters	9
1.13	Workbench Usage	15
1.14	Colors, Dithering and DISPLAY	16
1.15	Miscellaneous	18
1.16	Hall Of Fame	19

Chapter 1

JIV_e

1.1 JIV Documentation

JIV Documentation \$VER: JIV_e_guide 1.46 (13-Oct-96)

Copyright

Introduction

Distribution License

Shareware License

Installation

What's New?

Running JIV From A CLI Or SHELL

Running JIV From The Workbench

JIV's Parameters

Colors, Dithering and DISPLAY

Miscellaneous Notes

Acknowledgements

Please note: this document should be viewed using a non-proportional (mono-spaced) font, for example a Topaz or Courier font.

1.2 Author And Copyright

JIV program and documentation are © Copyright 1994-1996 by

Jürgen Weinelt
Zur Kanzel 1
D-97762 Hammelburg
Germany

jow@hcast.mayn.de
jow@rz.uni-wuerzburg.de

All rights reserved. JIV is provided "as-is" and subject to change without prior notice; no warranties are made. Use at your own risk. No liability or responsibility is assumed.

Please read the
distribution license
and the
shareware license
, too.

The JPEG codec is Copyright © 1994 Christoph Feck, TowerSystems. All Rights Reserved. It is based in part on the work of the Independent JPEG Group.

The JPEG codec is provided "AS-IS" and subject to change without prior notice; no warranties are made. All use is at your own risk. No liability or responsibility is assumed.

The Graphics Interchange Format(c) is the Copyright property of CompuServe Incorporated. GIF(sm) is a Service Mark property of CompuServe Incorporated.

The icons are © Copyright 1995 by Michael Hohmann and may not be re-used for other programs without his permission.

1.3 Introduction

Introduction

JIV is a picture viewer. I can imagine that you're wondering now why I had to write yet another one... well, here's a list of features that should answer your question:

- JIV supports many different
file formats
-
- JIV supports several types of
display hardware
 - JIV supports displaying over-sized images using autoscroll.
- JIV can automatically re-scale pictures.
- JIV can choose the right display mode automatically.

- JIV supports multiple filenames and AmigaDOS pattern matching.
- JIV can dither pictures that use more colors than your display hardware supports.
- JIV offers an endless/slideshow mode that can be used for (non-interactive) advertising, for example in a shop window.

Please note that JIV is probably not the best choice for "small" Amiga configurations. It will need a lot of memory and (in some cases) quite a bit of computing power as well.

The minimum configuration for JIV is:

- 68000 or better.
- AmigaOS 2.04 (V37) or better.
- 1 MByte or more of RAM.

This configuration should allow you to view small to medium-sized pictures (640x480x8 should just be possible).

The suggested configuration for JIV is:

- 68030 or better.
- AmigaOS 3.1 (V40) or better.
- 4 MBytes of RAM or more.
- Hard disk drive.

This configuration will allow you to view most pictures you're likely to find. A bit more RAM wouldn't hurt, though. :-) And of course you'll need a graphics board if you want to display true-color pictures without dithering.

1.4 Supported Image File Formats

Supported Image File Formats

- Both standard DataTypes and enhanced 24-bit DataTypes.
 - GIF (full GIF87a standard plus a subset of GIF89a).
 - BMP (MS-Windows and OS/2 variants, currently only uncompressed).
 - IFF-ILBM (including HAM6, HAM8, EHB and 24bit formats).
 - PNM pictures (PBM, PGM and PPM in both binary and ASCII format).
-

- PCX (8 bit, 24 bit, monochrome and `_some_` 4 bit).
- JPEG/JFIF, if the
 JPEG CODEC
 is installed.
- a reasonable subset of the TIFF 6.0 standard, including 1, 8 and 24 bit pictures in the compression types 1, 5 and 32773. JIV doesn't support the so-called "NeXT TIFF" format yet, but I'm working on the problems involved. Patience, please. :)

1.5 Supported Display Hardware

Supported Display Hardware

- all native graphics chipsets (OCS, ECS and AGA).
- Cybergraphics emulation.
- The A2024 greyscale monitor.
- Picasso-II/Village emulation.
- Generic mode, should work with every graphics board that emulates Workbench/Intuition. Tested with the Picasso-II emulation and Cybergraphics emulation. Beta-testers have successfully used JIV (with the Cybergraphics emulation) on Picasso-II, Spectrum and Retina boards.

1.6 Distribution

Distribution License

This version of JIV may be distributed freely for non-commercial purposes only. You must distribute the entire archive intact and unchanged, that is, you are not allowed to add files to the archive or remove files from the archive or change files in the archive.

Specifically, you're allowed to:

- Distribute JIV on floppy-disk-based collections of PD software.
 - Distribute JIV on CDROM-based collections of PD software in archived form only.
 - Distribute JIV electronically, for example through email or
-

FTP, or through BBS systems.

BBS sysops, do not add those pesky BBS advertisements to the archive!

Specifically, you're not allowed to:

- Distribute subsets of this archive or changed/patched versions of the files in this archive.
- Distribute JIV on CDROM in ready-to-run form, unless you obtain a specific license from the
author
first.
- Distribute JIV in any form on CDROMs that contain picture collections, unless you obtain a specific license from the
author
first.\$^1\$
- Distribute JIV bundled together with a commercial product (software, hardware or other), unless you obtain a specific license from the
author
first.

I have been informed that in the U.S. a patent has been granted for the LZW compression algorithm used in the GIF and TIFF file formats, and that the patent holder requires a license fee for use of this algorithm in commercial software (including shareware). Therefore I will not ask for any shareware fees from U.S. users of JIV.

Please do not confuse this distribution license, which details the terms and conditions for distributing JIV, with the

shareware license
, which details the terms and conditions
for using JIV.

\$^1\$ Permission is granted to include archived versions of JIV on the following CDROMs even if these CDROMs contain picture collections:

- Aminet
- Saar/AMOK

Angela Schmidt has my permission to install JIV in both archived and ready-to-run form on her Meeting-Pearls CDROMs.

Fred Fish has my permission to install JIV in both archived and ready-to-run form on his FreshFish CDROMs.

1.7 Shareware License

Shareware License

People keep telling me that the Shareware Concept, the way it was originally intended, is dead. They say you need to release crippled versions in order to get any response at all. They say you need to put annoying requesters, key files or even copy-protection code into your programs.

Consider this an experiment.

I'm putting a fully working version of JIV into circulation. Feel free to test it for as long as you want. If, and only if you're convinced that JIV does what you need and you'd like to keep using it indefinitely, pay what you think JIV is worth.

If I get the feeling that the experiment failed, I will do what the rest of the world does: create a crippled version for evaluation and a separate shareware version for registered users.

To encourage registrations, a shareware message has been added to this version of JIV. The message delays the program startup by approximately 1-2 seconds.

Remember, if you're using JIV, it can't be completely worthless...

It's up to you.

Here are a few examples of what I think might be appropriate:

- If you're just using JIV occasionally to view some pictures, how about paying 10 DM (equivalent of US\$5 to US\$10)?
- If you're using JIV regularly, I suggest paying 20-40 DM (equivalent of US\$15 to US\$25).
- If you're using JIV for commercial purposes (for example for commercial presentations), you might consider paying 50-60 DM (equivalent of US\$30 to US\$40).

Please note that "commercial use" in this context does not include CDROM distributions or other types of commercial distributions (see distribution license).

Remember, these are just suggestions. Feel free to donate whatever JIV is worth to you.

Send your donations to the
author
. For details,
please refer to "registration_e.txt".

Please do not confuse this shareware license, which details the
terms and conditions for using JIV, with the

distribution license
, which details the terms and conditions
for distributing JIV.

1.8 News

What's new?

- Several bugs were fixed.
- JIV supports both standard DataTypes and enhanced 24-bit DataTypes now.
- JIV can re-scale pictures that are smaller or larger than your display.
- JIV will add a unique error message number to each error message.

1.9 Installation

Installing JIV

It's simple.

Really.

Believe me. :-)

After unpacking the archive (you've already done that, otherwise you could not read this), just copy the JIV executable to a directory in your default search path. If you're using the Workbench, just drag the JIV icon into a drawer of your choice.

If you want to view JPEG images, you need to install the

JPEG CODEC
, too.

1.10 The JPEG CODEC

The JPEG CODEC And Datatype

In the subdirectory "JPEG", you'll find the JPEG codec and datatype. You need to install it if you want to view JPEG images. Please note that the datatype is not required, the CODEC is sufficient - that's important if you're using Kickstart 2.x, where datatypes are not supported at all.

To install the JPEG codec, double-click the Installer icon in the "JPEG" subdirectory.

The JPEG codec is Copyright © 1994 Christoph Feck, TowerSystems. All Rights Reserved. It is based in part on the work of the Independent JPEG Group.

The JPEG codec is provided "AS-IS" and subject to change without prior notice; no warranties are made. All use is at your own risk. No liability or responsibility is assumed.

1.11 SHELL Usage

Running JIV from a CLI or SHELL

To run JIV from a CLI or SHELL, just type "JIV" followed by its parameters.

The parameters are:

```
JIV F=FILE/M/A,   D=DELAY/K/N,   W=WAIT/K/N,   C=CENTER/S,
    Q=QUIET/S,    V=VERBOSE/S,   E=ERASE/S,   VERSION/S,
    DISPLAY/K,    WIDTH/K/N,    HEIGHT/K/N,  DEPTH/K/N,
    P=PROGRESS/S, ENDLESS/S,    S=SEQUENCE/K, H=HIDE/S,
    DITHER/K,    A=AUTOSCALE/K, Y=DATATYPES/K
```

Don't despair. The command line really is a bit long, but most of it is optional. If you simply want to view a picture, just type

```
JIV picture_name
```

and wait until the display opens. That's it.

For example, if you'd like to view the picture "neekid.gif", you type

```
JIV neekid.gif
```

at the shell prompt.

When the picture appears, JIV will expect you to press a

key or to click a mouse button to continue. Please note that oversized images will be displayed on an autoscroll screen, that is, you can scroll around just by moving the mouse pointer.

After playing around a bit, you'll probably want to use some of JIV's more advanced features. Here's the list of parameters with explanations. Please note that SHELL/CLI-parameters are not case sensitive.

Examples:

```
JIV display=a2024 center progress foo.bmp
```

Display "foo.bmp" on an A2024 monitor. Center it and add the progress indicator (which won't be too informative in this case).

```
JIV (f|g|h)#?.bmp delay=40 width=1024 height=768 progress autoscale=B
```

Display all the BMP pictures that start with "f", "g" or "h" on a 1024x768 display (note - this is only sensible if you own a graphics card like e.g. a Picasso-II). Display the progress indicator. Wait 4 seconds between pictures. Re-scale pictures if they are smaller or larger than 1024x768.

```
JIV display=native bla.gif
```

Caution. You should use "native" only if your Amiga has Kickstart 2.x and no graphics board.

You can interrupt JIV at any time by pressing CTRL-C (Control-C) in the Shell window JIV was started from. You can also press "q", "Q", "ESC" or "CTRL-C" on the image display screen when JIV is waiting for a mouse click or a key.

1.12 Program Parameters

JIV's Parameters

FILE The (list of) pictures that you want to view. You can specify every valid AmigaDOS filename pattern, for example "a#?.gif" to load all GIF images that start with an "a".

You must specify at least one filename.

FILE is not available as icon tool type (if you run JIV from the Workbench).

If JIV is started from the Workbench without any file parameters, it will now open a standard

ASL file requester. The requester supports multi-selection (shift key) as well as drag-and-drop operations.

DELAY specify the delay (in tenths of seconds) between subsequent pictures. This does not include the time needed to load and process the picture.

The default behaviour if you don't specify any delay is to wait indefinitely (until you press a key or click one of the mouse buttons).

Example: If you set DELAY to 55, JIV will load the first picture, then wait 5.5 seconds and then proceed to load the next picture (wait another 5.5 seconds, load the next picture etc. etc.)

Older versions of JIV expected the time to be specified in seconds.

JIV can be paused while displaying an image; just press "P" and the internal timer will stop. Press "P" a second time to re-start the timer.

Example usage: DELAY=64

WAIT GIF files can contain multiple pictures (slide shows or even small animations). WAIT specifies the time to wait between these pictures (in tenths of seconds).

The default behaviour if you do not specify any delay is not to wait at all.

Older versions of JIV expected the time to be specified in seconds.

Example usage: WAIT=18

CENTER center pictures on the screen; that is, if a picture is smaller than your screen, JIV will put it in the middle of your screen.

Please note that this feature may seem to fail under certain conditions; example:

(assuming a picture of 600x300 pixels)

```
JIV CENTER DISPLAY=GENERIC test.iff
```

will most likely seem to fail, because "generic" will open a standard AmigaDOS screen with exactly the same size as the picture (600x300). Because the screen is just as big as the picture, centering can not occur.

```
JIV CENTER DISPLAY=picasso test.iff
```

however will cause centering, because in Picasso mode,

screens will be restricted to certain dimensions. In this case, JIV will probably open a screen with 640x480 pixels and center the picture (with a left and right border of 20 pixels and a top and bottom border of 90 pixels).

If you want to force centering, you should also specify the screen dimensions using the WIDTH and HEIGHT parameters.

```
JIV CENTER WIDTH=640 HEIGHT=400 test.iff
```

will center the image regardless of the screen type.

Example usage: CENTER

QUIET causes JIV to suppress most of the messages that are usually written to the SHELL/CLI window (message window if started from Workbench). Only fatal error messages will be displayed.

Example usage: QUIET

VERBOSE causes JIV to display additional informations while loading pictures. VERBOSE implicitly overrides the QUIET flag.

Example usage: VERBOSE

ERASE causes JIV to erase the screen between subsequent pictures in multi-image files. See WAIT for an explanation of multi-image files.

Example usage: ERASE

VERSION displays version information plus acknowledgements for all parts of JIV. The practical value of this list is close to zero, but you may find the information interesting and/or amusing.

Note: You can not just call JIV with the VERSION parameter alone, because the FILE parameter is required even in this case. Let's call it a mis-feature. :-)

Example usage: VERSION

DISPLAY is used to explicitly specify the display hardware to be used for viewing pictures. Valid display types are currently:

AUTO This is the default.

In this mode, JIV attempts to guess the best possible display mode depending on your system configuration. It will choose

one of the following modes:

GENERIC, CYBER, PICASSO or NATIVE.

The A2024 mode will not be recognized and can therefore not be set automatically.

GENERIC This was the old default display.

In this mode, JIV will attempt to find a display mode depending on your system's capabilities.

For OS versions prior to 3.0 (V39), a fixed list of PAL and NTSC modes will be tested.

For OS versions 3.0 (V39), JIV will use the BestModeIDA() system call.

WARNING. If you're using an A4000/040 kickstart file on a computer with a CPU <= 68020, BestModeIDA() will crash your machine. This is not a bug in JIV.

NATIVE Will open an Amiga screen and then will write the image directly into the screen bitmap. If you're still using Kickstart 2.x, this is much faster than "GENERIC". It is not much faster with Kickstart 3.x, and you should never attempt to use it with foreign (GFX board) screens (it might even crash your Amiga).

PICASSO Open a Picasso screen (bypassing the Intuition emulation). You must specify this mode if you want to display true-color pictures.

Specifying DISPLAY=PICASSO if there is no Picasso-II board in your system (or if the Village intuition emulation is not running) will cause an error message.

You can't use DISPLAY=PICASSO if you've installed the Cybergraphics emulation.

Please read the "Miscellaneous Notes: Known Problems" section in this document.

CYBER Will explicitly ask for a Cybergraphics screen. You must specify this mode if you want to display true-color pictures.

Specifying DISPLAY=CYBER if the Cybergraphics emulation is not running will cause an error message.

A2024 Display the picture in an 8-color PAL mode screen on an A2024 monitor. This should (in theory) also work on NTSC systems, but it was only tested on PAL machines. Please contact me if there are problems with this display mode.

The A2024 mode will ignore WIDTH, HEIGHT and DEPTH.

Do not attempt to promote a 2024 display to a graphics board.

Example usage: DISPLAY=GENERIC

WIDTH forces a display width. For example, I'm running my Workbench on a 1024x768 screen (using a Picasso-II board). Switching to a different display resolution causes my monitor to re-synchronize, which in turn causes a nasty *click*ing from a relay inside the monitor.

To avoid this, I usually force JIV to open a display with the same resolution as my Workbench:

```
JIV WIDTH=1024 HEIGHT=768 DEPTH=8 .....
```

Example usage: WIDTH=640

HEIGHT forces a display height. See "WIDTH" for details.

Example usage: HEIGHT=480

DEPTH forces a display depth. Legal display depths depend on your display hardware:

OCS/ECS: 1...5 (2...32 colors)

AGA: 1...8 (2...256 colors)

PICASSO: In "DISPLAY=GENERIC" mode, 1...8 (2...256 colors)

In "DISPLAY=PICASSO" mode, 8, 16 and 24 (256, 65536 and 16 million colors, respectively).

In "DISPLAY=PICASSO" mode, you can also specify depths of 1...7, but you will simply get an 8 bit screen. It's a feature of the Picasso-II.

CYBER: same as PICASSO.

Example usage: DEPTH=4

PROGRESS displays some status information at the top of the

picture. This will be extended to a real progress indicator with a "fuel gauge"-style percentage display in future versions of JIV.

Example usage: PROGRESS

ENDLESS When all the pictures in the list of files have been displayed, JIV will normally end. If you specify ENDLESS, JIV will go to an endless loop displaying the pictures over and over again.

Example usage: ENDLESS

SEQUENCE JIV will normally display the pictures in the same order as they were specified on the command line (from Workbench, the order in which the picture icons were selected). With the SEQUENCE parameter, you can explicitly set the order to FORWARD (same as the default behaviour) or REVERSE (surprise - display the pictures in reverse order).

A future version of JIV may include additional SEQUENCE types (most likely a "RANDOM" option).

Example usage: SEQUENCE=REVERSE

HIDE JIV will normally load an image, then prepare a display in the background, then pop that screen to the front. With HIDE, you can prevent JIV from popping screens to the front (you'll have to do it manually).

Example usage: HIDE

DITHER There are two possible settings for the DITHER parameter:

QUICK JIV will use several shortcuts when it has to reduce the number of colors for displaying an image on your display hardware. The kind of shortcuts used depends on the kind of reduction needed, but generally, QUICK will result in poor picture quality and should only be used for getting a quick preview.

Exception: With DISPLAY=A2024, you may want to use DITHER=QUICK for displaying line drawings or pictures with lots of text in them.

RANDOM JIV will use high-quality algorithms for color reduction; dithering will be done using random dithering.

This is the default.

Example usage: DITHER=QUICK

AUTOSCALE This parameter controls the behaviour of the automatic re-scaling feature:

N turns off the re-scaling entirely.
U enlarges pictures that are too small.
D shrinks pictures that are too large.
B both enlarges and shrinks pictures as necessary.

The default is "D".

Example usage: AUTOSCALE=N

DATATYPES This parameter controls the behaviour of the DataTypes loader function:

OFF turns off the DataTypes loader entirely.
FIRST attempts to load a picture through DataTypes before attempting to load it through the internal loaders.
LAST attempts to load a picture through the internal loaders before attempting to load it through DataTypes.

The default is "LAST".

Example usage: DATATYPES=OFF

1.13 Workbench Usage

Running JIV from the Workbench

There are three ways of using JIV from the Workbench:

1 Default tool

Specify JIV as the default tool of an image icon and add image-specific parameters as tool types to the image icon. Then double-click the image icon to view the picture.

2 Multi-selection

You can also use Workbench's multi-selection feature to view pictures using JIV. Click (single-click) on an image icon, then press the shift key on your keyboard and continue to single-click as many image icons as you wish. When you're done, double click on the JIV program icon (while still holding the shift key).

3 If JIV is started from the Workbench without any file parameters, it will now open a standard ASL file requester. The requester supports multi-selection (shift key) as well as drag-and-drop operations.

In any case, you can specify default parameters as tool types to the JIV program icon, and specify image-specific parameters as tool types to the individual image icons.

The parameters are basically the same as on the SHELL/CLI command line, with two major exceptions: tool types are case sensitive, and you can not specify FILE parameters as tool types.

When JIV is started from the Workbench, it opens a window where you can enter "CTRL-C" (Control-C) at any time if you wish to interrupt JIV.

Example tool types:

(In the JIV program icon)

```
DISPLAY=A2024
CENTER
PROGRESS
```

Default to displaying pictures on an A2024 monitor, centering them and showing the progress indicator.

(In an image icon)

```
CENTER
WIDTH=1024
HEIGHT=768
PROGRESS
AUTOSCALE=B
```

Override the CENTER, WIDTH, HEIGHT, PROGRESS, and AUTOSCALE defaults of the JIV program icon.

1.14 Colors, Dithering and DISPLAY

Colors, Dithering and DISPLAY

Color display and dithering depend on your graphics hardware, as well as the settings of the DISPLAY and DEPTH parameters. DISPLAY and DEPTH will also interact with the WIDTH and HEIGHT settings.

Here are a few examples (SHELL syntax):

(all the examples assume a 640x480 color JPG picture. Color JPG implies 24bit truecolor, BTW).

- JIV DISPLAY=GENERIC DEPTH=8 test.jpg

will fail, unless there is AGA or supported third-party display hardware available in your system. Other GFX boards with pseudo-conformant Intuition emulations may work, too. On AGA, you will get an 8-bit-screen; resolution depends on your Monitor type configuration (for example, NTSC 640x400x8, PAL 640x480x8 or Multiscan Prod. 640x480x8). These exact resolutions may also depend on your Overscan prefs. On a Picasso-II, you will get a 640x480x8 screen.

- JIV DISPLAY=GENERIC DEPTH=5 test.jpg

will dither the truecolor image using a set of 32 colors. This setting should work on all the supported graphics hardware (except A2024 of course). On OCS and ECS Amigas you will get a 320x200 (NTSC) or 320x256 (PAL) screen with 32 colors; on AGA the resolution will depend on your Monitor prefs (you could also use DEPTH of up to 8 on AGA). On a Picasso-II, you will get a 640x480 screen.

- JIV DISPLAY=GENERIC DEPTH=4 test.jpg

will dither the truecolor image using a set of 16 colors. This setting should work on all the supported graphics hardware (except A2024 of course). On ECS, OCS and AGA, you will get a 640x400 (NTSC) or 640x480 (PAL) screen with 16 colors (you could also use DEPTH=8 on AGA); the AGA resolutions can again vary depending on your Monitor prefs. On a Picasso-II, you will get a 640x480 screen.

- JIV DISPLAY=PICASSO test.jpg

will get you a truecolor display on a Picasso-II board.

- JIV DISPLAY=PICASSO DEPTH=8 test.jpg

will dither the image using a 256 color palette on a Picasso-II board.

- JIV DISPLAY=PICASSO DEPTH=5 test.jpg

will dither the image using a 256 color palette on a Picasso-II board. This is a special case, because if you set DISPLAY=PICASSO, DEPTH settings of less than "8" will still open 256-color screens. :-)

- JIV DISPLAY=PICASSO DEPTH=16 test.jpeg

will display the image using a 65536 (5x6x5) color space. This is rather close to true color, and no dithering will be done.

- All the DISPLAY=PICASSO examples are valid for DISPLAY=CYBER, too.

- IF JIV fails with an error message, you might want run it again adding VERBOSE to the parameter list. With VERBOSE, most error messages will be somewhat longer.

1.15 Miscellaneous

Miscellaneous Notes

Todo:

- support for additional display devices (Retina?)
- support for additional image file formats:
 - Icon
 - XBM
 - Targa
 - others
- GUI
- AREXX port
- Image manipulation and format conversion.

Known problems:

- The VERSION parameter can not be specified alone. You need to specify an image file name, too. Call it "mis-feature", O.K.? :-)
- If you're using an A4000/040 kickstart file on a computer with a CPU \leq 68020, the DISPLAY=GENERIC may crash your machine. This is not a bug in JIV. The BestModeIDA() system call in the A4000 kickfile seems to contain an instruction that is only available on 68030 and above.

As far as I know, there are only two kinds of people who use A4000 kickstart files on CPUs below 68030: (former) registered developers and software pirates.

- If you click the left mouse button to end viewing a picture, you should move the mouse pointer a bit down. Right at the top of the display, there's an invisible drag bar so you can still drag your screen up and down.
 - The "DISPLAY=GENERIC" mode will be very slow if you're using older versions of AmigaOS (before 3.0/V39). JIV uses WritePixelLine8() in GENERIC mode; WritePixelLine8() is much faster in AmigaOS 3.0 (V39) and later than in the older versions. If you're still using such an old Kickstart version, you might want to use "DISPLAY=NATIVE".
 - If you're a user of the Cybergraphics emulation, please
-

make sure that version 40.45 (or later) of it is installed on your Amiga. With older versions, you may run into problems when displaying oversized images.

- If you press the "P" key while JIV is displaying a picture, JIV may seem to "freeze" (in fact you just activated the "PAUSE" mode). Just press "P" again in order to re-start it.

1.16 Hall Of Fame

Acknowledgements

The following people contributed their ideas, software or support to JIV. In alphabetical order:

Christoph Feck,	he wrote the JPEG codec and helped by patiently answering my questions.
Marc Heuler,	beta-tester.
Arne Hinrichsen,	beta-tester.
Michael Hohmann,	he made suggestions, tested several beta versions and was kind enough not to insist on playing "Spaceward HO!" every other weekend. :-) He also designed those nice icons.
Christian Hombach,	beta-tester.
Markus Illenseer,	beta-tester.
Frank Mariak,	he patiently answered Picasso-II and CyberGraphics questions.
Bernhard Möllemann,	beta-tester and other help.
Carsten Orthbandt,	beta-tester.
Henning Sauer,	beta-tester.
Matthias Scheler,	he contributed the asynchronous I/O routines and helped in the beta-testing and debugging.
Stefan Sell,	he tested several betas to the limit.

Whithout you, JIV would not exist. Thank you!

The following people contributed bug reports:

Joel A. Corn,	he found the "PNM header parser" bug that was fixed in JIVPNM 1.5.
---------------	--

Christian Rattei, he found several omissions in the
IFF code.

Niels Van Willigenburg, he reported several bugs, especially
in the BMP and TIFF code.

Thank you!
