jpegAGA

COLLABORATORS

	TITLE : jpegAGA		
ACTION	*NAME*	*DATE*	*SIGNATURE*
WRITTEN BY		December 12, 2024	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

Chapter 1

jpegAGA

1.1 jpegAGA 2.2

```
                        jpegAGA 2.2 by Günther Röhrich
                        ******************************
```

jpegAGA is, at this time, the highest quality JPEG viewer. (if a map-file for that picture is available) To use it you need the AGA chipset, a 68020 or higher processor and OS 3.0 or higher. This program uses the HAM8-mode, grayscale display will use 256-color-mode. Although jpegAGA 2.x has many new features it is still fully compatible to the previous 1.x versions. In this release version 6 of the Independent JPEG Group's JPEG software is used for decoding.

If a precomputed colormap (a file with the ending .map) is available then the picture will be displayed in a higher quality compared to all other viewers.

With the supplied shell script you can create such map-files for all your pictures. These files will only be 198 bytes long.

If a map-file is not available then you will get a lower quality display. For grayscale display this file is not needed.

Table of contents

1.2 contents

Table of contents
****

Picture screen
History
Announcement
Address of the author

1.3 advantages

```
The advantages of jpegAGA:
**************************

- very high display quality (better than any other viewer)
- complete source is available
- best compatibility to all JPEG pictures because the unmodified JPEG source
  from the Independent JPEG group is used for decoding
  (jpegAGA 2.2 uses version 6)
- support for grayscale JPEG’s and for showing color pictures in grayscale
  mode
- automatic screen mode selection, depending on the size of the picture
- pictures can be shown at reduced size, useful for high speed previews
- they can be scaled automatically to fit completely on your screen
- when more than one picture is selected jpegAGA will load and decode the next
  picture in the background while you are still looking at the previous one
- supports progressive JPEG’s
```

1.4 thanks

```
Copyright notes and expression of thanks
****************************************

This program and the supplied source is Public Domain, you  can  do  what  you
want with it. For the IJG JPEG sources please read the file README.


I have to thank the following persons/groups:

My betatesters:
 Denis Ahrens (d.ahrens@bamp.berlinet.in-berlin.de)
 Lars Tewes (LSR@quick.sh.sub.de)
 Tim Reddy (tcreddy@prairienet.org)
and all the others not mentioned here.

Gunther Nikl (gnikl@informatik.uni-rostock.de) for his help ans suggestions

Pawel Hernik (spi-ph@srv1.tu.kielce.pl) and others for the  chunky  to  planar
 source code

The Independent JPEG Group (jpeg-info@uunet.uu.net):
 for the source to decode/encode the JPEG picture format

Free Software Foundation, Inc. and all contributors for the GNU C compiler and
related tools
```

Martin Huttenloher (xen@magic.in-ulm.de)
 for the wonderful MagicWB icons

1.5 installation

Installation
****

Copy the programs from the bin directory anywhere into your shell's search path. The file name extensions are showing for what processors they are compiled. After installation you should remove these extensions. Workbench users can place the main program (jpegAGA.020) anywhere they want.

```
Extension 000:  68000 and 68010 processors, runs on all others too
Extension 020:  only for 68020 or higher processors
Extension 020881: only for 68020 or higher processors with a coprocessor
```

All programs are compiled with the gcc 2.7.0 compiler. You don't need the ixemul.library any more.

NOTE: jpegAGA 2.2 does not work any more on gfx cards. It now uses a custom, incompatible, chunky to planar converter.

For using the shell scripts CreateMap(Big) you have to assign JPEGTMP: to a directory on your hard disk with enough free space. (e.g. assign JPEGTMP: HD:T) The programs djpeg and cjpeg will eventually need this directory too. jpegAGA itself may also need this directory when processing large progressive JPEG's. The best solution is to add this command to the user-startup file.

The shell scripts CreateMap and CreateMapBig should be copied to the S: directory. They assume that the programs djpeg and ppm2AGA are available (with exactly these names) in the shell's search path and the JPEGTMP: directory exists.

1.6 invocation

Invoking jpegAGA:

from the Shell:

```
  jpegAGA options inputfiles
```

from the Workbench:

Double click on the program icon or on a project icon of your pictures. Of course, you have to prepare the project icons first.

Shell-users can use the script PV for fast previews. It is used similar as jpegAGA itself but lot of options are predefined. Of course, you can create your own Shell scripts with prefered options. This saves you a lot of work at typing in the shell.

Inputfiles
****

The name of the JPEG-file that should be read. You can specify as many files as you want, at any place in the command line. Wildcards (e.g. #?.jpg) are also supported. If you want to use a file name that starts with "-" you must use "?" instead. (Otherwise jpegAGA will assume that you used a wrong option.) If you have not specified any file names a file requester will be displayed. You can select multiple pictures with the Shift key. Remove the default pattern if you want to select file names that do not end with ".jpg" or ".jpeg".

If you started the program with a double click from Workbench you will also get the file requester if you have not selected any pictures. (by using the Shift key) It is much more comfortable when you create a project icon for each of your pictures. Enter jpegAGA as the default tool, including the complete path. Now you can look at the picture simply by double clicking on it. You can select multiple pictures with the extended selection feature of the Workbench. This is recommended because jpegAGA will then use background loading and economize time.

New in version 2.2: Progressive JPEG's are now supported. It can take a while from the start of reading such files till the screen is opened, so don't worry. If the file is in progressive mode then it will be indicated by printing "progr." in the console window.

Grayscale pictures will be shown on a 256 colors screen without extensive computations. Unfortunately there are also grayscale pictures encoded in color mode. You should use the -GRAY option for such pictures. (And send a complaint to the person who created the picture.)

When showing the picture in color mode the map-file for this picture will be used if it is available. The map-file must have the same name as the picture with the additional ending .map. To support the PC-filesystem a name that has the extension replaced by ".map" will also be accepted. If a map-file is found this will be notified in the console window.

Examples for file names:

```
picture file: picture.jpeg    map-file: picture.jpeg.map   or  picture.map
              pic1.jpg        map-file: pic1.jpg.map       or  pic1.map
```

jpegAGA will also try to find the map-file in the directory pointed to by the environment variable MAPDIR.

1.7 options

Options
****

Options can be used anywhere in the command line, they are case-insensitive. If they are not set then internal default values will be used.

Workbench users can set options with the "ARGS" tool type.

(e.g. ARGS= -vga -scale 1/2 -gray)
If the picture has a icon then the tool types of this icon will also be taken into account. These option will override those from the jpegAGA program icon, but they are individual, only for a single picture. If you selected the picture with the file requester the tool types of the pictures will not be processed. This is also the case for Shell invocation.

-GRAY

Display picture in grayscale mode. This is much faster than color display and is therefore useful for a fast preview.

-VGA

The picture will always use the VGA screenmode (also called ’multiscan’) for display. Use this option when you know that the picture was intended for displaying on a PC or other machines with square screen resolution. (this is the case for most JPEG-pictures)

NOTE: With ’square’ I don’t mean that the screen has the same number of pixels in horizontal and vertical direction. (e.g. 1024x1024) I mean that a square composed of e.g. 100x100 pixels will be shown as a square on the screen. Because a usual picture tube has a ratio of 4 to 3 the screen’s resolution must be a multiple of that. (e.g. 640x480 because

```
 640   4
 --- = - )
 480   3
```

-SUPER72

This option will only be accepted if you also set the -VGA option. The SUPER72 screen mode will be used for very large pictures. You have to modify the SUPER72 monitor driver in most of the cases (e.g. with the program "moned") if you want to get a square resolution in that mode.

-SCALE 1/x (new in version 2.0)

The picture will be scaled by a factor of 1/x. Supported are 1/1, 1/2, 1/4 and 1/8. Pictures at reduced size will be shown much faster, you can use this option for a fast preview.

-FIT (new in version 2.0)

The picture will be scaled to fit completely on the screen. A tolerance of about 35 pixels is allowed. Scaling is possible only with the factors 1/2, 1/4 and 1/8. The options -VGA and -SUPER72 will also be taken into account. The SUPER72 mode will only be used if the picture is too big for a VGA screen.

-NOSMOOTH (new in version 2.0)

-DCT FAST (new in version 2.0)

With these options decoding is a bit faster but the picture quality gets worse. You can use these options for fast previews.

-SMR (new in version 2.0)

When this option is set you can choose the screen mode with a requester. Background loading does not work any more because you must choose a screen mode before a picture is shown. This option is ignored if it is set as a tool type of a picture icon.

-MAXMEM N (new in version 2.2)

This option sets the maximum amount of memory that should be used for decoding progressive JPEG files. All the DCT coefficients for the whole picture must be stored. The default setting is 1MB. N must be set in kilobytes.
(e.g. -maxmem 200 for 200K)

Add the ending M for megabytes. (e.g. -maxmem 4M for 4 megabytes) If more memory is needed then jpegAGA will use temporary files in the JPEGTMP: directory. Of course, these will be deleted automatically.

Alternatively, you may use the JPEGMEM environment variable. It works in the same manner.

For picture icons there are additional options available. You can use them to disable options that are set in the jpegAGA icon. For example, if you use the option -NOGRAY in the picture icon then the picture will be shown colored, even if the -GRAY option is set in the jpegAGA icon.

option	counterpart (picture icon only)
-DCT FAST	-DCT INT
-NOSMOOTH	-SMOOTH
-GRAY	-NOGRAY
-VGA	-NOVGA
-SUPER72	-NOSUPER72
-FIT	-NOFIT

If you started jpegAGA with a wrong option (e.g. -v) you will get a list of all available options, including a short description. This works from Shell only.

1.8 programs

There is also a JPEG decompression program included. It converts JPEG pictures to the ppm format. (It is also needed for creating the map-files.) To use it enter:

```
 djpeg inputfile >outputfile
```

In some cases djpeg will try to store temporary files to JPEGTMP:, you have then to assign this directory to somewhere on your hard disk. (for example assign JPEGTMP: HD:T) For more informations read the included documentation. With the program cjpeg you can create your own JPEG files.

1.9 map

Creating the map-files
****

These files are needed for showing color picture with best possible quality. Normally, they are created by the shell-script CreateMap. Usage: (from Shell)

```
 CreateMap picturefile
```

After succesfull completion a map-file with the correct name will be created. (On PC-filesystems too) You can process several pictures with a single invocation. The command

```
 SPat CreateMap pictures/#?.jpg
```

will create, for example, a map-file for all pictures with the extension .jpg in the directory pictures. If it is not possible to store the map-file at the same location as the picture itself then it will be stored in the directory pointed to by the environment variable MAPDIR. You can set the environment variable e.g. with the command:

```
 setenv MAPDIR HD:tempdir
```

If you reset your machine the variable will get lost. To prevent this you can use the command:

```
 copy ENV:MAPDIR ENVARC:
```

Now you can also show pictures from a CD-ROM with best possible quality.

NOTE: You must take care that the same map-file will not be used for different pictures. Otherwise you will get lower picture quality.

With the shell-script "CreateMapBig" mapfile creation is much faster. While decoding the picture it will be scaled down by 1/2. This does no harm (at least for large pictures) because only the color distribution is of interest and not the whole picture. For very big pictures you may even scale by 1/4, you will have to edit the script a bit.

For small pictures (e.g. 320x200 pixels) you should not reduce the size and use the normal "CreateMap" script. It is also recommended for use if you are processing a complete directory with both big and small pictures.

If problems with map-file creation occur then please check the following:

- Have you installed everything as it is described in the section "Installation" ?

The files djpeg and ppm2AGA are needed under these names in the search path. The shell-script itself must be in the S: directory.

```
  Note: - ppm2AGA must have version 1.4 or higher
        - use djpeg from this distribution only
```

- Have you assigned the directory JPEGTMP: correctly ?

- Has the hard disk partion enough free space ?

  When using the CreateMapBig script less disk space will be needed.

- Is enough memory available ?

If not you can reduce memory usage by using the -M0 option with the shell script. (See the documentation for ppm2AGA for more.)

```
  example: CreateMap pic.jpg -M0 or SPat CreateMap #?.jpg -M0
```

- Is the disk, on which the JPEG-picture is stored, write enabled and does it has enough free space ?

- Have you set the environment variable MAPDIR correctly ? You can check this with the command: getenv MAPDIR

You can interrupt the shell-script at any time by pressing CTRL-C. In this case all temporary files in the directory JPEGTMP: will be removed automatically.

1.10 screen

The picture screen
****

You can move it around with the mouse. It also has an invisible, but full functional drag bar and depth gadget. In color mode you will see errors at the left side, they are caused by the HAM encoding. In this version of jpegAGA they are smaller than with all other programs. Pressing the right mouse button will close the screen or switch to the next one. If there is not enough Chip RAM to open the next screen in the background jpegAGA will wait till the previous screen will get closed and will try it again. If the mouse pointer remains in the busy state after a picture is completely finished then you will know that the next one is loaded in the background. Pressing the left mouse button has no effect. This means that you can activate another screen at any time and re-activate the picture screen again. Pressing the Esc key will completely finish showing the set of pictures.

NOTE: Because of some bugs in the OS it can happen that pictures are shown cropped or even totally corrupted. In this case you should move the picture to the right till it is shown correctly.

1.11 history

```
History
*******

1.0 - 4-June-1994

    - first public release

1.1 - 5-October-1994

    - added the option -SUPER72

    - the picture screen no longer flashes when closing it and  the  delay  is
      much smaller now

    - map-files will also be searched in  the  directory  pointed  to  by  the
      MAPDIR environment variable

    - if a map-file is not found the user will get informed that he/she should
      create one

    - file/path names of any length are now accepted

2.0 - 14-January-1995

    - slightly faster

    - the ixemul.library is not needed any more

    - version 5A of the JPEG source from the Independent JPEG Group is used
      (instead of V4 in jpegAGA 1.x)

    - input buffer size is now 32768 bytes, normal dos.library calls  will  be
      used for all file operations

    - added the options -SCALE, -FIT, -NOSMOOTH, -DCT FAST and -SMR

    - added the background loading feature

    - pressing the Esc button will completely stop the program,  pressing  the
      right mouse button switches to the next picture

    - the picture screen opens with the  correct  colors,  no  flashing  after
      opening

    - multiple files and wildcards are supported, at any place in the  command
      line

    - selecting the pictures with a file requester is possible

    - full Workbench support, but there is still no GUI

    - messages in the console window are much better

    - there is now a makefile for the Aztec compiler supplied with the source
```

- enhanced the HAM8 encoder, if you move the picture screen around the errors on the left side will be much smaller

2.1 - 4-February-1995

- bug fixed: when the window or the temporary bitmap could not be opened on a screen the program will operate now correctly

2.2 - 17-September-1995

- version 6 of the JPEG source from the Independent JPEG Group is used

- compiled with gcc 2.7.0 and libnix 1.0

- progressive JPEG files are supported

- file requester appears again when all selected pictures are shown

- requesters now reappear with their previous settings

- a custom chunky to planar routine is used, this makes screen rendering much faster (20% faster in color mode, 50% in grayscale mode)

- the program reacts on pressing CTRL-C in the console window (not always)

- new option and environment variable: -maxmem and JPEGMEM (needed to support progressive JPEG's

- documentation is now in AmigaGuide format

1.12 announcement

Announcement
****

My spare time is very limited because I am now studying Electrical Engineering at the University Stuttgart. Therefore I am searching for other developers to cooperate with me and beta-testers for testing purposes. I am especially searching for someone to write a GUI. If you want to join in then please contact me. I will then create a detailed description of the source. You can use any C development system available for the Amiga but only the gcc compiler will create the best possible optimization.

To recompile ppm2AGA you need the newiff package from Commodore. V37 is available on Fish-disk 705, V39 on the "3.1 Amiga Developer Update" disks and on Aminet.

For compiling the programs you will need the OS 3.x includes. They are available in the "3.1 Amiga Developer Update" or on the Fresh Fish CD-ROM's.

1.13 author

If you have additional questions, suggestions for enhancements, bug reports, Mungwall/Enforcer hits, etc. then please contact me. It would also be nice if you would send me a corrected version of this manual. (and for ppm2AGA too) To contact me write to:

```
  electronic mail:                                    normal mail:
  ****************                                    ************

 Guenther@studbox.uni-stuttgart.de                    Günther Röhrich
     (Internet EMAIL)                                 Lerchenbergstr. 4
                                                      73733 Esslingen
                                                          Germany
```

You may also visit my personal WWW homepage at:

http://rpool1.rus.uni-stuttgart.de/~etk10325/