

**AutoPEG**

**COLLABORATORS**

	<i>TITLE :</i> AutoPEG		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		March 29, 2022	

**REVISION HISTORY**

NUMBER	DATE	DESCRIPTION	NAME

# Contents

<b>1</b>	<b>AutoPEG</b>	<b>1</b>
1.1	Welcome to AutoPeg . . . . .	1
1.2	Please read this! . . . . .	1
1.3	Installing AutoPEG . . . . .	2
1.4	What's this? . . . . .	2
1.5	What to do... . . . .	3
1.6	Useful Examples . . . . .	4
1.7	Using Scripts . . . . .	5
1.8	Compatilby . . . . .	6
1.9	Watch out for... . . . .	6
1.10	Call me! . . . . .	7

---

# Chapter 1

## AutoPEG

### 1.1 Welcome to AutoPeg

AutoPEG

Automatic JPEG decompression for the Amiga

by John Corigliano

**Legal Stuff** Notices and acknowledgements.

**Install** Installing AutoPEG.

**Intro** What is AutoPEG.

**Usage** How to use AutoPEG.

**MetaUsage** Advanced usage.

**Test 1-2-3** Programs I've tested.

**Warning** Things to watch out for!

**Contact** How to contact the author.

### 1.2 Please read this!

--<<\*< AutoPEG is copyrighted © 1994 by John Corigliano.

--<<\*< AutoPEG is released as Gift Ware - there is no obligation to register this software, but hey, if you want to send me something, I won't refuse ;) Acceptable payment can be anything from a simple E-Mail message to a new car!

--<<\*< This archive may be FREELY redistributed, as long as it has not been modified.

--<<\*< AutoPEG may not be used as part of or included in any commercial product without the author's consent.

--<<\*< AutoPEG is offered AS IS. No warranties or guarantees are offered on this product or its effect on life, the universe, you know, everything.

---

--<<\*< AutoPEG would not have been possible if not for the work of Thomas G. Lane and the Independent JPEG Group. Thanks, guys and gals.

--<<\*< Also, many thanks to the authors of the NetPBM software, especially 'ppmto24'.

--<<\*< Finally, thanks to Michael Saunby, who's HAM8-JPEG code was an essential example of implementing the JPEG code.

--<<\*< AutoPEG source code (in C and Assembler) is available on request.

### 1.3 Installing AutoPEG

I would advice running the included "AutoPEG.Install"\* program. If you choose not to, here's what needs to be done:

Copy "AutoPEG" to any drawer.

If you have a 68000, 68010, or 68020 copy the "AutoPEG.exe" file to C:

If you have a 68030, or 68040+ copy the "AutoPEG.exe.030" file to C: and rename it as "AutoPEG.exe"

The "AutoPEG.exe" program does the JPEG decompression. I made it a seperate file to save your RAM: - with a tradeoff in speed. AutoPEG.exe can be made resident.

\* Note: to run the "AutoPEG.Install" program you need the "Installer" program (which was written by CBM, but never included in the OS!).

You most likely have it somewhere (if you use WB 3.0 you can find it on your "Install Disk"). Many programs use "Installer" so, if I were you, I would find it and copy it to C: right now!

### 1.4 What's this?

Welcome to AutoPEG

AutoPEG is a small utility that allows programs that can access 24-bit image files access to compressed JPEG files. This is done in a manner that is totally transparent to the program. Here's how it works:

When a program tries to open a file AutoPEG intercepts the request and checks to see if the file to be opened is a JPEG file. If it is not, then AutoPEG ignores the request and sends it on its way. If it is a JPEG file, it is renamed, then decompressed as a 24-bit file with the original name. Simple, yet elegant!

The fuel guage - Whenever AutoPEG decompresses a file, it opens a small window on the active screen. In this window is a "fuel guage" that lets you know that AutoPEG is at work. This is mostly for psychological reasons, as JPEG decompression can take from 30 seconds to two or three minutes or more, depending on the size of the file and the speed of your system.

---

## 1.5 What to do...

### Starting AutoPEG

AutoPEG can be run from CLI or WorkBench (if run from CLI, it will detach). If AutoPEG is to be run from the WBStartup drawer, DO NOT delete the DONOTWAIT tooltype.

To start AutoPEG run it from CLI or WorkBench. To remove AutoPEG run it again.

AutoPEG takes four arguments which may be entered from the command line or as tooltypes (they MUST BE capitalized):

QUIET=TRUE

QUIET=FALSE

- Normally, AutoPEG posts a requester to let you know that it has just loaded or it has just quit. If you use the QUIET=TRUE argument, it will not post these requesters.

USEMEM=TRUE

USEMEM=FALSE

- By default, AutoPEG will decompress JPEG files directly to disk. This saves memory but is very slow (depending on disk speed). If you have vast quantities of free memory, you can specify USEMEM=TRUE and AutoPEG will decompress JPEG files to a file called "Ram:T/AUPG" and then copy that file to disk (using 4K blocks). This is much faster but requires that you have the free ram to do so.

ON=TRUE

ON=FALSE

- If AutoPEG is running and you run it again, it will terminate. However, if you want AutoPEG to load, but aren't sure if it's already loaded, use this argument. Thus, if you have AutoPEG running and run it again with the ON=TRUE arg, it will not terminate, but rather, stay in memory.

OFF=TRUE

OFF=FALSE

- This is very similar to ON except that it forces AutoPEG to remove itself if it is running. Or it does nothing if AutoPEG is not running.

\* These arguments are very useful when running AutoPEG from **scripts**.

## 1.6 Useful Examples

This section shows two examples of using AutoPEG via DOS scripts:

1. Using AutoPEG only for a specific program.

Let's say I've decided that I only want to use AutoPEG when I use a program called 3D-Atomizer. What I would do is launch AutoPEG and the 3D-Atomizer program from a script. Let's call this script "3d.script". (Let's assume I have AutoPEG in Sys:Utilities and 3D-Atomizer in Work:Gfx/Trace) To do this, I'll open up my favorite text editor and enter:

```
; Script to launch AutoPEG and 3D-Atomizer
```

```
Sys:Utilities/AutoPEG QUIET=TRUE ON=TRUE
```

```
Work:Gfx/Trace/3D-Atomizer
```

```
Sys:Utilities/AutoPEG QUIET=TRUE OFF=TRUE
```

I'll **save this script** as Work:3d.script.

Here's what the script does:

Line #1: is a comment

Line #2: runs AutoPEG, with no requesters, and forces it to load if it isn't loaded, or stay loaded if it is

Line #3: runs the 3D-Atomizer program

Line #4: removes AutoPEG, with no requesters, and forces it to remove itself if it is running, or forces it to not load if it isn't running

This script ensures that the AutoPEG program will only be running while the 3D-Atomizer program is running.

2. Using AutoPEG all the time except for a specific program.

Let's pretend that I have AutoPEG load everytime I turn on my Amiga (via the user-startup script). Also, let's say that I have a program called SuperPaint that has automatic JPEG decompression built in. Since SuperPaint's JPEG loader will most likely be faster than using AutoPEG I want to remove AutoPEG from memory before I launch the SuperPaint program. Let's make things more interesting by assuming that SuperPaint needs a "SuperPaint:" assign.

If SuperPaint is in Work:Gfx/2D then our script would be:

```
; Script to launch SuperPaint without AutoPEG
```

```
Sys:Utilities/AutoPEG QUIET=TRUE OFF=TRUE
```

```
assign SuperPaint: Work:Gfx/2D
```

```
SuperPaint:SuperPaint
```

```
assign SuperPaint: REMOVE
```

Sys:Utilities/AutoPEG QUIET=TRUE ON=TRUE

I'll **save this script** as Work:super.script.

Here's what the script does:

Line #1: is a comment

Line #2: forces AutoPEG out of memory

Line #3: makes the necessary assign

Line #4: runs the SuperPaint program

Line #5: removes the SuperPaint: assign

Line #6: forces AutoPEG back into memory

This script ensures that the AutoPEG program will not be running while the Superpaint program is running.

## 1.7 Using Scripts

Running scripts from CLI

All you need to do to use a script via a CLI is to launch it using the "execute" command. To launch a script called "my.script" enter:

```
execute my.script
```

at any CLI prompt.

Running scripts from WorkBench:

To use a script via WorkBench you need to create an icon for the script with the "IconEdit" program (in sys:tools"). The icon must be a Project icon, so select the Type->Project menu item.

Create the image and save it: if the script is Work:my.script, save the icon as Work:my.script.info.

Next, from the WorkBench, select the icon by clicking on it one time, and select the Icons->Information... menu item from the WorkBench menu. When the window opens, set the default tool to c:iconx

Then click the Save gadget. IconX is a program that runs Amiga DOS scripts from the WorkBench, something they cannot do on their own.

When you launch the script, by double-clicking its icon, you will notice that a window appears. This is IconX's output window. If the program you are launching doesn't have any text output, you can eliminate this window by selecting the icon, selecting the Icons->Information menu item from the WorkBench menu, and adding this tooltype:

```
CON=NIL:
```

(If this really intrigues you check out your Amiga DOS manual for information on creating console windows).

---

## 1.8 Compatibility

Okay, so my arsenal of programs that support 24-bit images isn't exactly enormous, but here's how my programs faired with AutoPEG installed:

Program Comaptible? Notes

Real3D v1.4 Yes! R3D loads textures into memory before each render (instead of only once).

Slightly annoying.

Aladdin4D No. I think it is incompatible because AutoPEG doesn't write CAMG chunks to the 24-bit IFF. Not a problem, though because A4D has internal JPEG support.

ViewTek Yes! ViewTek reads each file twice - way slow! Okay, though because it has internal JPEG support.

DCTV Paint Yes! Smokin'!

## 1.9 Watch out for...

DO NOT attempt to access JPEG files on a CD-ROM with AutoPEG! The reason is that the disk that contains the file must be Read/Write and CD-ROMs are ReadOnly. This is because when a program tries to open (for example) "Work:Pix/huge.jpg" it is returned a lock on "Work:Pix/Huge.jpg", though, unbeknownst to it, the file is decompressed. If I get any complaints about this, maybe I'll change it. I know this is a problem, but I'm working on it.

I wouldn't reccomend copying JPEG files when AutoPEG is loaded. I tried it, and it didn't work! However, it is okay to copy non-JPEG files.

AutoPEG currently doesn't support automatic compression of IFF files.

Always try to terminate a program normally. Don't just shut you computer off if a program that you are running has accessed a JPEG file. You see, AutoPEG renames the original JPEG file, then decompresses it as a 24-bit file (with the original file's name). Then, when the file is closed by the program that opened it, the 24-bit file is deleted and the JPEG file is renamed back to its original name. Thus, if a program has an open JPEG file and you shut the computer off before it closes the file, then the next time you turn your computer on you will have two files instead of one - the original (albeit renamed) JPEG file and the 24-bit file. Just between you and me, this won't usually be a problem as most programs open the file, read the contents, then close it right away.

There is (currently) a maximum of 50 JPEG files that can be open at one time.

---

## 1.10 Call me!

Please feel free to send me E-mail to report bugs (yechhh!), or suggest improvements, or just to say "Hello!"

InfoService User-ID Internet address

GENie j.corigliano j.corigliano@genie.geis.com\*

Bix mopp mopp@bix.com

Delphi j\_mopp j\_moop@delphi.com

\*I prefer GENie simply because I have a nice mail reader for it!

Snail Mail: John Corigliano

1118 Rodman Rd.

Wilmington, DE 19805

U.S.A.

If you would like the source code, you might first try to get these files (FTP from Aminet):

pub/aminet/gfx/show/JPEGv42source.lha - source for JPEG

pub/aminet/gfx/pbm/netpbm1.lha - source for PBM

pub/aminet/gfx/show/ham8\_jpeg.lha - example of using JPEG

Or, if you like, I can E-Mail you my modified source.

---