

ImageToAVI CL Documentation

Table of Contents:

- 1) [How to purchase this software](#)
- 2) [List of commands and switches.](#)
- 3) [How to create an AVI or GIF animation.](#)
- 4) [How to load images from an input file.](#)
- 5) [How to apply effects to pictures \(not yet supported in CLI version 1.0\).](#)
- 6) [2GB file size limit.](#)
- 7) [How to extract pictures from an AVI video \(not yet supported in CLI version 1.0\).](#)
- 8) [Miscellaneous](#)
- 9) [How to contact ASW Software](#)

**A picture is worth a thousand words.
So how many words is your slideshow worth?**



How to purchase ImageToAVI

We thank you trying this software. We strive to create programs that work and will continue doing our best to create quality software for low prices. If you should find any errors in this product, please report them to ImageToAVI@aswsoftware.com. You may find product updates at www.ASWsoftware.com. Updates are free for one year from time of purchase, or for five minor version upgrades (i.e. version 1.5 to version 2.0), whichever one is later.

To purchase ImageToAVI, please **visit the website where this software was downloaded** so that a commission can be paid to our affiliates. If you can not remember where you downloaded it, visit: www.aswsoftware.com or see contact information at the end of this help file.

List of Commands and Switches

Example1: `ImgToAVI [/p [path to pics]] [output name] [-w [width]] [-h [height]]
[/fps [framesPerSecond]]`

Example2: `ImgToAVI [/t [text file]] [output name] [-w [width]] [-h [height]]`

Required Parameters in order of priority:

`/p [folder path]`.....Use folder containing list of pictures.
Not required if `/t` is used.
`/t [filename]`.....Use text file containing a return delimited path to
each picture. Not required if `/p` is used.
`[output name]`.....Example `c:\output.avi` or `output.avi` (relative path)
Existing file will be overwritten by default.

Options (order not important):

`-w [width]`.....Output video width (integer). Defaults to the
width of the first image in the sorted list.
`-h [height]`.....Output video height (integer) Defaults to the
height of the first image in the sorted list.
`-subf`.....Include sub folders in input path. Valid only if `/p` is used.
`-fps [FPS]`.....Output frames per second (float). Defaults to 29.97
`-rt [value]`.....Repeat each input frame n times (float). Defaults to 1.
`-ds [seconds]`.....Display each input frame f seconds (float).
Defaults to value of `-rt` divided by value of `-fps`.
`-ia [filename]`.....Include PCM WAV audio file (for AVI only).
`-gif [loops] [t] [d]`.Output as GIF animation, instead of default AVI.
All 3 options after `-gif` are required.
`-[loops]` (integer), 0 = infinity.
`-[t]` is transparency type (integer), 0 = not
transparent, 1 = top left pixel, 2 = bottom left pixel.
`-[d]` is draw method, 0 = Draw each frame new, 1 =
Draw on top of previous frame.
`-sf`.....Sort list image file names before processing.
`-sb`.....Same as `sf`, but sorts backwards.
`-sd [depth]`.....Sort depth (integer). Only the last N chars are
compared in the sort. Defaults to -1, unlimited depth.
`-ar`.....Do not maintain aspect ratio.
`-zi`.....Do not fit smaller images to display size.
`-bsw [%]`.....Border size width % (float). Defaults to zero.
`-bsh [%]`.....Border size height % (float). Defaults to zero.
`-pf [value]`.....Pixel format (8 = 8bit, 24 = 24bit).
Defaults to 24bit.
`-rm [value]`.....Resize method (1=standard, 2=smooth, 3=bilinear). Defaults to 3.
`-bgc [r] [g] [b]`.....Background color (integers 0-255).
Defaults to 0 0 0 (black).
`-bgi [filename]`.....Include background image. Stretches to fit.
`-of`.....Open output file when finished. (Plays AVI or GIF
in default player when finished)
`-lf [fileName]`.....Generate log file. Note:
-"Exit - Success" will be outputted on the last line if
successful.
-"Exit - Warnings" followed by the number of warnings will be
outputted if warnings occurred.
-"Exit - Aborted" will be outputted if failed.
-Overwrites existing log file.
`-co`.....Cancels operation if file "Abort.ita" exists in the
output directory. Deletes this file after aborted.

-hcd [FourCC].....Hide codec dialog. Requires a FourCC code string.
-Example1: "-hcd DIB" for uncompressed AVI.
-Example2: "-hcd XVID" for XVideo AVI.
-Example3: "-hcd FFDS" for ffdshow video codec pack.
-Example4: "-hcd CVID" for Cinepak video codec.
-The fourCC is case sensitive and must not exceed 4 characters.

-mi [value].....Maximum image count (integer). Defaults to 100,000

-hew.....Hide error windows. This option will output error messages to the console and log file only.

-owp.....Turn on overwrite protection. A confirmation window will be shown if output file already exists.

-fdir.....Force create output directories. If the output folder path does not exist, the path will be created.

-sem [maskList].....Set extension mask list. Used only when loading pics from a folder. The default mask list includes all supported images. Must be entered as a comma delimited string of extension masks, no spaces.
-Example: -sem *.jpg,*.jpeg,*.bmp

Supported Formats: (Output: Standard AVI, GIF animation)
(Input: JPEG, BMP, GIF, ICO, WMF, EMF, WAV (PCM format))

How to create an AVI or GIF animation

This software can load images either from a file, or all pictures from a specified path. See the section titled "[List of commands and switches](#)" for command details. See section title "[How to load images from an input file](#)" for those details.

For an AVI:

Example 1 (with windowed compression options displayed, to run behind windowed GUI):

Command: "imgtoavi /p Bckgrnds output.avi -subf -w 350 -h 200 -fps 15.0 -ds 0.2 -bgc 20 100 255 -sf -lf log.log -co -of"

Explanation:

Creates an AVI using all the pictures found in the Bckgrnds directory, including sub folders (due to presence of -subf switch). Width and height are 350x200. Frames per second is 15. Each image to be displayed for 0.2 seconds. Background color is light blue. Images will be sorted first. A log file will be created. The presence of switch -co will cause program to quit if a file called 'Abort.ita' is created in the same folder as ImageToAVI. The -of causes file to be opened and played with default player when finished.

NOTE: If you are loading images contained in the same directory as ImageToAVI, then use double quotes with a dot in the middle: "."

Example 2 (with windowed compression options hidden, to run with compression options already specified):

Command: "imgtoavi /p "Bckgrnds\paths with a space" output.avi -subf -w 350 -h 200 -fps 15.0 -ds 0.2 -bgc 20 100 255 -sf -lf log.log -co -of -hcd cvid -hew"

Explanation:

Same as example 1, except the compression options window will not be shown, and all errors, except fatal errors, will be outputted to console and log file. The cinpack codec will be used. Also, the folder location of the images to load includes quotes to show an example of how to load file paths that contain spaces. If you are loading images contained in the same directory as ImageToAVI, then use double quotes with a dot in the middle: "."

NOTE: If overwrite protection is set to on, then a window will still be shown to prompt for permission to overwrite existing file.

NOTE2: A valid FOURCC code must be used and the codec must be installed on your system. You can get valid FOURCC codes by doing an internet search, or by checking the included codec's list in file 'FOURCC Codes.pdf'.

Example 3 (with windowed compression options displayed, to run behind windowed GUI):

Command: "imgtoavi /t exinput.txt output.avi -w 350 -h 200 -fps 2.0 -ds 0.5 -bgc 20 100 255 -sf -lf log.log -co -of"

Explanation:

Creates an AVI using all the pictures found in the 'ExInput.txt' file. Width and height are 350x200. Frames per second is 2. Each image to be displayed for 0.5 seconds. Background color is light blue. Images will be sorted first. A log file will be created. The presence of switch -co will cause program to quit if a file called 'Abort.ita' is created in the same folder as ImageToAVI.

For an animated GIF:

Same as AVI, except, use these additional commands: -gif [loops] [t] [d]

All 3 options after -gif are required.

-[loops] (integer), 0 = infinity.

-[t] is transparency type (integer), 0 = not transparent, 1 = top left pixel, 2 = bottom left pixel.

-[d] is draw method, 0 = Draw each frame new, 1 = Draw on top of previous frame.

NOTE: Not all browsers or GIF animation players will play at the correct frame rate. Usually they play slower than the selected frame rate. Some play right on. Microsoft applications tend to play slower.

Example 1 (with error messages outputted to log file and console):

Command: "imgtoavi /p "Bckgrnds\pics to use" output.gif -subf -w 350 -h 200 -fps 15.0 -ds 0.2 -bgc 20 100 255 -sf -lf log.log -co -of -hew -gif 0 0 0"

Explanation:

Creates an animated GIF using all the pictures found in the input directory, including sub folders (due to presence of -subf switch). Width and height are 350x200. Frames per second is 15. Each image to be displayed for 0.2 seconds. Background color is light blue. Images will be sorted first. A log file will be created. The presence of switch -co will cause program to quit if a file called 'Abort.ita' is created in the same folder as ImageToAVI. The -of causes file to be opened and played with default player when finished.

NOTE: If you are loading images contained in the same directory as ImageToAVI, then use double quotes with a dot in the middle: "."

How to load images from an input file

Step 1: Create an 'input.txt' file and paste everything between the red dashes into it:

```
0 //The very first entry in this file must be the number of images to load.  
//This number must be followed by a space or return character.
```

```
/****** Instructions for loading from a file *****/
```

There are two ways to load images into the CLI version of ImageToAVI. The first is to load the images from a folder. The second is to load the images from a file such as this one. Follow the below instructions to load from a file.

Instructions: The number of images in the list must be on the very first line of this file. Each file path must be contained between double quotes. Each image does not have to be followed by a space or return character.

For example: A full path: "C:\output\test.avi"
A relative path: "output\test.gif"

Image formats supported: JPEG, BMP, GIF, ICO, WMF, EMF.
Reserved symbols (outside of " "): <> ? * /\ : | ^ % &
To make ImageToAVI ignore a single line, use: //
To ignore a section of text, use: /* ignored text here */

```
/******
```

```
//Put images below here
```

Second: Change the 0 on the first line to the number of images that you will use.

Third:

Add the names of the pictures, including their paths with each path between double quotes, such as:

```
"Bckgrnds\Man Made\Motorcycle001.JPG"  
"Bckgrnds\Man Made\MouseLaser001.JPG"
```

Fourth:

Run ImgToAVI with /t command instead of /p. See examples in section titled "[How to create an AVI or GIF animation.](#)"

NOTE: You don't need all those instructions in your input file. All that is required is that you have a space or a return after the number of images and that the image paths be separated by double quotes, for example:

```
2  
"Bckgrnds\Man Made\Motorcycle001.JPG"  
"Bckgrnds\Man Made\MouseLaser001.JPG"
```

How to apply effects to pictures

(Not yet supported in the command line version)

2GB file size limit

(See NOTE below for a possible work around for this limitation)

Due to the limitations of the standard AVI format, AVI files created that are greater than 2GB will not function properly if at all. This limit was brought about by the geniuses at Microsoft. I give them a 10 for smart thinking. Why doesn't this program stop you from creating a file bigger than 2GB's? Because the genius here at ASW software doesn't want to think about that right now.

There is a new AVI format out there called OpenDML or AVI 2.0. It can have a maximum file size that is so big that a 1000GB hard drive wouldn't even come close to holding it. We hope to add AVI 2.0 functionality in this program in a future release.

(ohh ya, when I said I give Microsoft a 10 for smart thinking, I meant on a scale of 10 to 100)



NOTE: Some users have found that you can create an AVI over 2GB if you load the finished AVI into a video conversion program called VirtualDub, and then re-save it. This will convert it to OpenDML.

How to extract pictures from an AVI video

(Not yet supported in the command line version)

Miscellaneous

- This version was designed to run under a GUI, and therefore will default to windows being shown for user information or input. This can be defeated by using the commands: **-hew** and **-hcd** if you wish to limit user interaction.
- Currently **only PCM wav** audio is supported.
- AVI duration will be longer than the estimated video length if the audio duration is longer than the video length.
- Using some codec's will cause either an error, or no file to be created. If you get an '**EAccess Violation**' error, try changing the resolution of the output AVI. XVID's MPEG-4 compressor seems to have trouble with non-standard resolutions. Try something like 720 x 480 or 800 x 600. If a file is not created at all, then use a different codec all together. Those are usually compatible with OpenDML, which is not supported by this version.
- Other options will soon be added to ImageToAVI (like transitions, more effects and batch file converters).

How to contact ASW Software

Contact name: Anthony West

Questions or comments: ImageToAVI_CL@aswsoftware.com or aswsoftware@mail.com.

For updates visit www.aswsoftware.com

**A picture is worth a thousand words.
So how many words is your slideshow worth?**

