

# Image'n'Bits Help Index

To choose a topic press Tab to select the underlined topic you want to view, and then press Enter. With the mouse, point to the underlined topic you want to view, and then click the mouse button.

If you aren't  
going to read  
anything else,  
please read  
**THIS !**

[Introduction](#)  
[Payment and Legalese](#)  
[Program History](#)  
[This Release Features](#)

[Quick Glossary](#)

## **Menus**

[File Menu Commands](#)  
[Edit Menu Commands](#)  
[View Menu Commands](#)  
[Image Menu Commands](#)  
[Options Menu Commands](#)  
[Window Menu Commands](#)

## **Glossary**

*1 bit image* - an image that is capable of having a maximum of 2 colors (typically black and white, but not necessarily).

*4 bit image* - an image that is capable of having a maximum of 16 colors.

*8 bit image* - an image that is capable of having a maximum of 256 colors.

*24 bit or "true color"* - an image that is capable of having a maximum of 16.7 million colors.

*Compound document* - a document which serves as a container for diverse data created by different applications. Image'n'Bits documents are capable of storing not only images, but also various data created by "foreign" applications. See [OleClient](#).

## Important Miscellaneous

### Performance

Needless to say, the more powerful the hardware, the better. There is one particular image type/hardware combination which deserves noting: when displaying 24 bit (16 million or true color) images on a 256 color Windows setup, Windows must match every pixel in the image against the available 256 colors. This process is computationally very expensive. Such a computation takes place every time an image is moved or overlapped by another window. It is a problem even on a 33/66 MHz 486 with a Local Bus graphics card. The solution to this problem is as follows: First display the image. Experiment with [View palettes](#) to select the best "image look" (this will also require some time). [Convert your 24 bit image to a 256 color](#) image. Finally, close the 24 bit image and continue working with the new 256 color image.

### Color

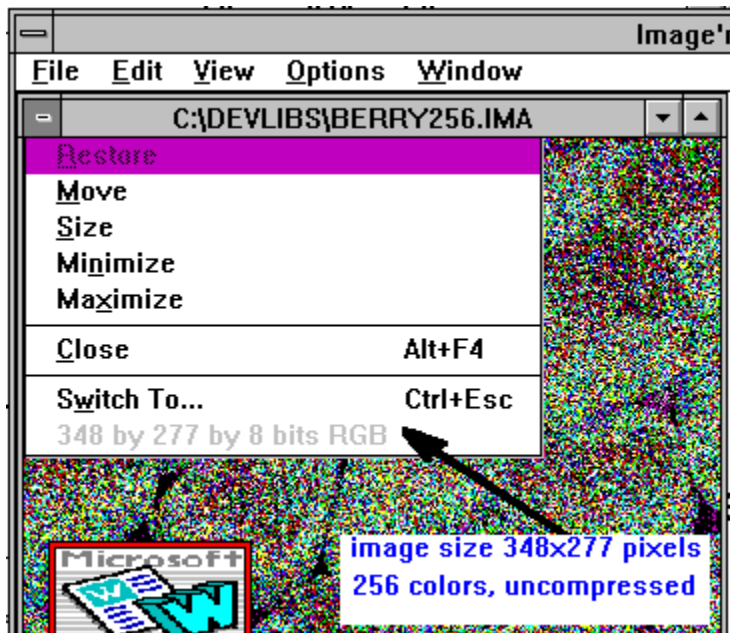
Image'n'Bits does a very good job of preserving the image color contents regardless of the number of colors available on your Windows system. This means that if the image you are working on is a 16 million color image, all the color data is retained by the image even though your Windows setup is a 16 color VGA. Although this image may not look like anything it is supposed to be (after all, the system only has 16 colors), you can apply various Image'n'Bits operations to improve the image's appearance on your system. The exceptions to this "device independence" rule are:

[Edit Paste](#)

[Paste From File](#)

[Add Text](#)

After you perform one of the above operations your image will become device dependent, i.e. it will contain the same number of colors as your Windows system. This is fine while you continue to work with the image on the screen, but if you decide to print the image on the color printer you will get no more colors than what you saw on your PC. To check the color content of your image, click on the window's system menu.



I know how nobody likes to read the documentation, but Image'n'Bits is full of imaging

related terminology and concepts. It's for your own benefit to read the rest of this help file. Thank you for reading this much.

**Thanks**

to all beta testers who contributed to and tested this product.

Special thanks to [Albert Duro](#) for his insights and ideas and for really going the extra mile on this product.

## Introduction

Welcome and thank you for using Image'n'Bits. This first release of the product is a first step in the development of what will become a very comprehensive imaging program.

What's so special about Image'n'Bits?

**It preserves image color content regardless of your hardware configuration.** Let's say you have a "true" color (16.7 million colors) TIFF file format image which you would like to give to your friend. Unfortunately, your friend does not have any TIFF viewers, but he does have a PCX viewer. So you are going to convert the image from TIFF to PCX file format on your 16 color VGA Windows system. How many colors should the converted image have? I say it should still be 16.7 million colors even though the conversion process took place on a 16 color system. Most users will agree with this approach and it's kind of obvious, but many commercial Windows imaging/desktop products would "loose" the color content during the conversion process and limit the output image to the number of colors available on the hardware. Why? It's the easier and therefore much shorter application development process (from Windows application development point of view). As a user, you would never know the difference on a 16 color computer, but you would if you printed it on a color printer or took the new image to a PC with better color capabilities.

**Commercial application features at Shareware price.** [See Payment and Legal](#)

**Image'n'Bits was developed and tested under Debug version of Windows** which means fewer (if any) Unrecoverable Application Errors (unfortunately, many Windows programmers have never even heard of Debug flavor of MS-Windows).

**Image'n'Bits imaging smarts are residing in Dynamic Link Libraries and can be used for the development of other programs.** Contact [Bananas Software Inc](#) for licensing and pricing information.

## **The First Release**

This first release of Image'n'Bits includes the following features:

Simultaneous multiple image display

Import and Export of various image file formats

Printing

Full Clipboard support

Monitor Gamma Correction

Bitmap Format Conversions (1, 4, 8, and 24 bits)

Edge Detection, Blurring, Sharpening, Tuning and Special Effects Algorithms

Drag and Drop

OLE Client Linking and Embedding

## **Program History**

Image'n'Bits started out as a test program for a collection of various imaging libraries. As the libraries grew in size and complexity, so did the test program. Gradually, the focus of development had shifted from just imaging libraries to making Image'n'Bits an end user product: the user interface was revamped, file management, OLE, printing, Clipboard, and Drag and Drop capabilities were added and voila! The first release of a product is born. I welcome user 'wish list' suggestions, enhancements, and bug reports.

## Payment and Legal

This program is not free. This software (Image'n'Bits version 1.0 unregistered) is **Shareware**. You are allowed to copy it, distribute it, upload it to bulletin boards and use it. You may not, however, change or alter the program files in any way. The program is property of Bananas Software Inc. You may not resell or repackage it without the author's permission. The DLLs that make up Image'n'Bits product can be used only WITH Image'n'Bits product and should not be used with any other product for any reason and in any way. The name Image'n'Bits is copyrighted by Bananas Software Inc. and is a trademark of Bananas Software Inc.

Bananas Software Inc. and Alex Lerner are not in any way liable for any damages incurred by the use of this software. The user assumes full responsibility for the use of this software.

Bananas Software Inc. is a privately owned company formed in 1985 by Alex Lerner. We specialize in MS-Windows and Presentation Manager development and consulting. The author of this program, Alex Lerner, can be contacted through Compuserve (70372,3466) or the U.S. mail service.

Bananas Software, Inc.  
c/o Alex Lerner  
29 Engle Rd  
Paramus, NJ 07652

If you like the program and would like to show your support and appreciation by becoming a registered user, send me your name, address and \$25 in U.S. currency. I also welcome user features 'wish list', enhancements, suggestions and bug reports.



## File Menu Commands

New

Open

Save

Save As

Print Setup

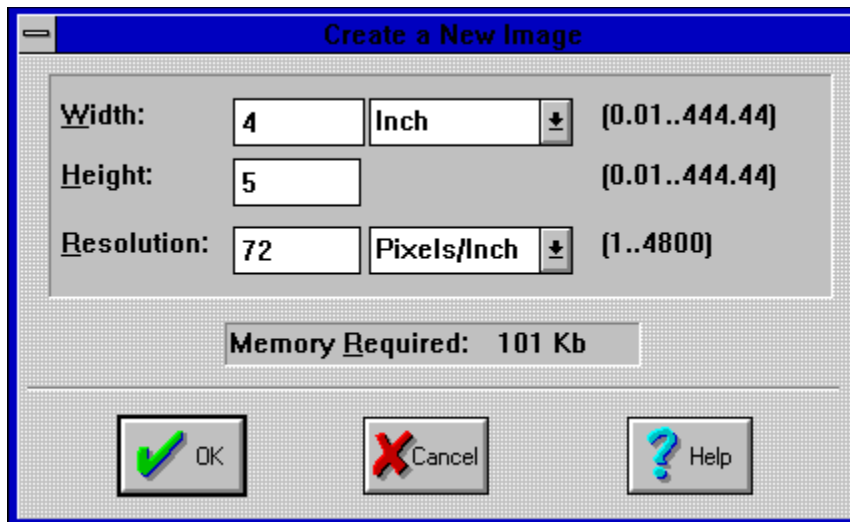
Print

Import

Export

Exit

## File New Menu Command



**New** dialog box creates a new document with a "blank" image in it.

The "blank" image size is important only if you are planning to do Clipboard Paste and Paste From File operations since the the "blank" image will determine the boundaries of the pasted clippings.

When importing an image the size of the blank image is not important since it gets entirely replaced by the newly imported image.

## **File Open Menu Command**

**Open** dialog box opens existing Image'n'Bits compound documents (IMA file extension) and images. All opened images are automatically converted to the Image'n'Bits compound document format and are appended with the IMA file extension instead of the original image file extension. For example, bitmap file koala.bmp will be opened into koala.ima document window.

**See Also**  
[File Import](#)

## **File Save Menu Command**

**Save** menu command saves Image'n'Bits document under its already existing filename. When saving the document for the very first time a Save As dialog box will come up prompting you for the document file name.

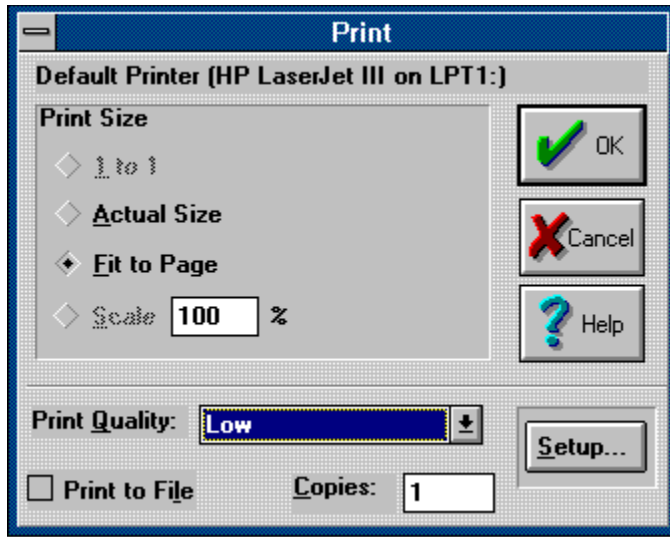
## **File Save As Menu Command**

**Save As** menu command saves Image'n'Bits document under a new filename. The document on the screen will also get renamed.

## **File Print Setup Menu Command**

**Print Setup** menu command brings up Print Setup Dialog Box which allows you to make the printer selection and various printing attributes such as paper orientation, source of paper, etc.

## File Print Menu Command



**Print** menu selection prints the image inside the Image'n'Bits document. The OLE objects inside the document do not get printed. The size of the printed image depends on the Print Dialog Box **Print Size** setting. The Print Size settings available in the first release of Image'n'Bits are **Actual Size** and **Fit To Page**.

## File Import Menu Command

**Import** menu command is very similar to the **Open** menu command in that it too opens files. **Import** menu command brings different file format images into an already open Image'n'Bits document. The image inside the existing Image'n'Bits document is replaced with the incoming image. The OLE objects inside the open document are unaffected by this operation.

### **The first release of Image'n'Bits can import the following file formats:**

MS-Windows bitmaps: all flavors (BMP, DIB, RLE file extensions)

MS-Windows metafiles: Aldus Placeable and Windows metafiles (WMF file extension)

MS-Windows Clipboard: CLP file extension

DKB Trace Dump Files: DIS file extension

Compuserve GIF files: GIF file extension

**More file formats are coming!** The following file formats are in the works: Targa, TIFF, PCX, Kodak CD Format and JPEG. Help me to finish the development of these new exciting formats by registering your copy of Image'n'Bits.



## **File Export Menu Command**

**Export** menu command is a way to save an image in a particular file format, other than Image'n'Bits proprietary IMA format.

### **The first release of Image'n'Bits can export the following file formats:**

MS-Windows bitmaps: all flavors (BMP, DIB, RLE file extensions).

Compuserve GIF files: GIF file extension

MS-Windows metafiles: WMF file extension

**More file formats are coming!** The following file formats are in the works: Targa, TIFF, PCX, Kodak CD Format and JPEG. Help me to finish the development of these new exciting formats by [registering](#) your copy of Image'n'Bits.

## **Edit Menu Commands**

### **Clipboard Commands**

Cut

Copy

Paste

Paste Link

Clear Command

### **OLE Commands**

Links

Insert Object

Paste From File

In addition to the above commands, Edit Menu Popup displays **Object** menu command as the last entry when a document is open. **Object** menu command becomes ~~OLE object~~ specific when an OLE object is selected within the document. For example, when an Excel spreadsheet OLE object is embedded within the Image'n'Bits document and is currently selected, **Object** menu command would transform itself into Edit Excel Spreadsheet Object. Selecting that menu command would then bring up Excel with the embedded object already in it ready for editing and update.

## **Image Menu Commands**

Flip Horizontal  
Flip Vertical

Histogram

### **Tune Commands**

Gamma Correction  
Color Adjustment  
Posterize

### **Convert To Commands**

Line Art  
16 Colors  
256 Colors  
24 Bit ColorDither  
Gray Scale  
Negative Image/Solarize

### **Smoothing Filters**

Blur  
Blur More  
Major Blur

### **Sharpening Filters**

Sharpen  
Sharpen More  
Real Sharp  
Edge Enhancement  
Find Edge  
Find Edge and Invert  
More Edge Enhancement

### **Special Filters**

Minimum  
Median  
Maximum  
Gradient Edge Detection  
Embossing

### **Funhouse Filters**

Wavy  
Fisheye  
Spiral  
Pinch  
Melting

Add TextCombine Images

## **Image Posterize Menu Command**

Image **Posterize** menu command reduces the number of colors in the image. The possible number of colors varies from 8 (1 bit per channel) to 2 million (7 bits per channel).

### **See Also**

[Affected Image Area](#)

## Image Combine Images Menu Command

Image **Combine Images** menu command allows you to combine images in different ways. The images (or selected areas within an image) do not have to be of the same size. Here are the rules for figuring out the destination image's affected area:

- images get operated on starting from the upper left corner. The size of the new image is equal to the size of the image from the *destination images* column.
- when a portion of an image is selected using the selection rectangle., only that portion of the image is affected. If both source and destination images have selection rectangles, the affected area size of the destination image is equal to the smaller of the two rectangles.

The operations supported are:

- addition
- subtraction
- averaging
- minimum
- maximum
- overlay (copy)

- logical AND, OR, and exclusive OR
- multiplication
- division

*Addition* through *overlay* operations are very useful "classical" image processing operations. The rest of the operations were added for "completeness" and are not useful for anything but unusual special effects.

### **See Also**

[Affected Image Area](#)

## **Image Minimum, Median, Maximum Menu Commands**

**Minimum** image filter emphasizes dark colors of the image.

**Median** image filter emphasizes the image's mid-tones. This filter can also be used to clean up single stray pixels which showed up in the image as a result of the scanning noise.

**Maximum** image filter emphasizes highlights of the image.

### **See Also**

[Affected Image Area](#)

## **Image Melting Menu Command**

Melting operation, as the name implies, gives an image the appearance of being melted. You can melt the image vertically (downward) or horizontally (to the right), one RGB color channel at a time or all three channels simultaneously (default). Melting 1 channel is faster than melting all three and melting a grayscale image is faster than melting a color image.

When you are melting a grayscale image, mark it as such using *Image is a grayscale image* button. This will speed up the melting process.

### **See Also**

[Affected Image Area](#)

## Image More Edge Enhancement Menu Command

**More edge enhancement** dialog box offers additional edge detection algorithms. The Sobel edge detection algorithm is visually the most interesting of the three since it allows you to overlay the "detected" edges on top of the original image.

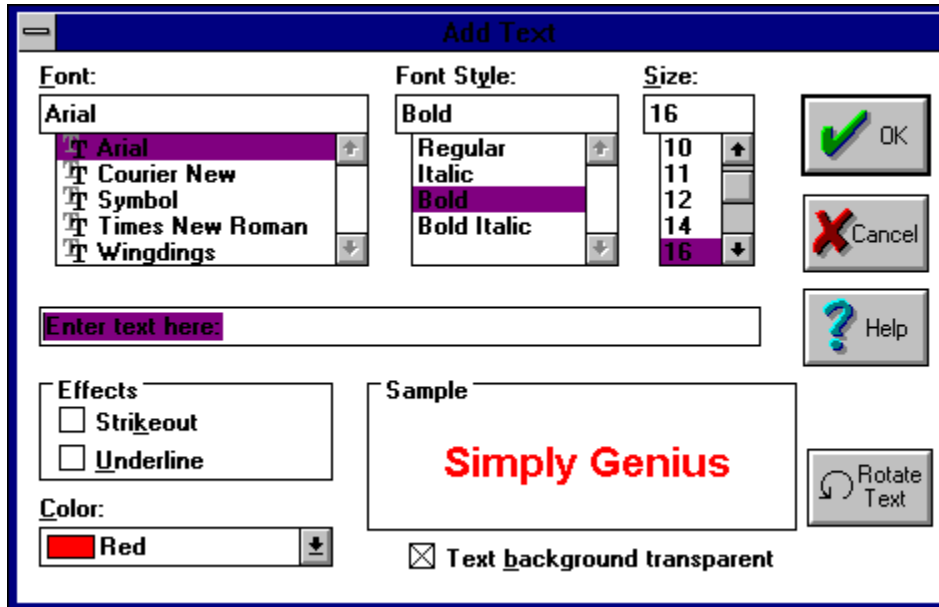
### **See Also**

[Affected Image Area](#)



## Image Add Text Menu Selection

**Add Text** dialog box displays various text attributes. In addition to the font, size and text style selections the dialog box also contains *Text background transparent* and *Rotate Text* buttons.



*Text background transparent* is turned on by default. When this option is on, the background image is visible through the text. When this option is turned off, text is placed on the image with black color as background.

### Note:

- **Add Text** dialog box lists True Type fonts only.
- When *Text background transparent* option is checked, text cannot be fitted into the selection rectangle. Text will be placed on the image as if the selection rectangle was not defined.
- When rotating text, portions of the first and last letters can get clipped. If that is the case, pad text using spaces.

## **Edit Cut and Edit Copy Menu Commands**

**Copy** menu command copies a portion of the image (if a portion is selected) or an entire image (if nothing is selected) to the clipboard. **Cut** menu selection does the same thing as **Copy**, but in addition to copying it also deletes the portion of the image that was just copied. Essentially, Image'n'Bits Cut and Copy work like any other Cut and Copy.

### **See Also**

[Edit Paste](#) and [Edit Paste Link](#)

## Edit Paste Menu Command

**Paste** menu command pastes (copies) a selection that you defined using **Edit Copy** or **Edit Cut** menu commands from the clipboard into your document.

**Paste** menu selection works differently depending on the clipboard contents and the lack or presence of the selection rectangle on the image you are pasting into. If the clipboard contents can be pasted as a bitmap (image), they will be pasted as such. If the selection rectangle is defined on the image, the new clipping will be fitted into the selection rectangle stretching or shrinking the clipping as necessary. This applies to bitmap clippings only. If a selection rectangle is not defined, the clipping will appear in the upper left corner of the image. In either case, the clipping can be dragged to its final destination by pressing the left mouse button on top of the clipping. To cancel the paste operation, press right mouse button (or press the escape key). To complete and accept the paste operation double click on the clipping using the left mouse button. While the paste operation is in progress all menu selections will be disabled including closing of the document window in which the image is displayed.

If the clipboard contents can not be pasted as a bitmap, they will be pasted as an OLE object.

**Note:** Performing the paste operation on an image makes that image device dependent. For example, you are working on a 24 bit image on a 256 color system. You have copied one part of the image and pasted it on to another area of the image. If you check the image window's system menu you will see that the image you are working on has become a 256 color image. You shouldn't care since your system is only 256 colors anyway. On the other hand, you can apply a richer set of operations to a 24 bit image. Also, when printing to a color printer, your printout will contain only 256 colors.

**See Also**  
Affected Image Area

## **OLE Objects**

Image'n'Bits is an OLE Client. Image'n'Bits allows the user to place "foreign" objects (supported and understood by the applications that created them) right into the Image'n'Bits documents. An OLE object can contain practically anything: from sound to live video, from bitmaps to text.

OLE objects are placed into Image'n'Bits documents through Paste, Paste Link, and Insert Object menu commands. There are two flavors of OLE objects: Embedded and Linked.

## **Edit Paste Link Menu Command**

**Edit Paste Link** command places linked OLE objects inside the Image'n'Bits compound document. See [Embedded OLE objects](#) and [Linked OLE Objects](#).

### **See also**

[Edit Insert Object](#)

[Edit Links](#)

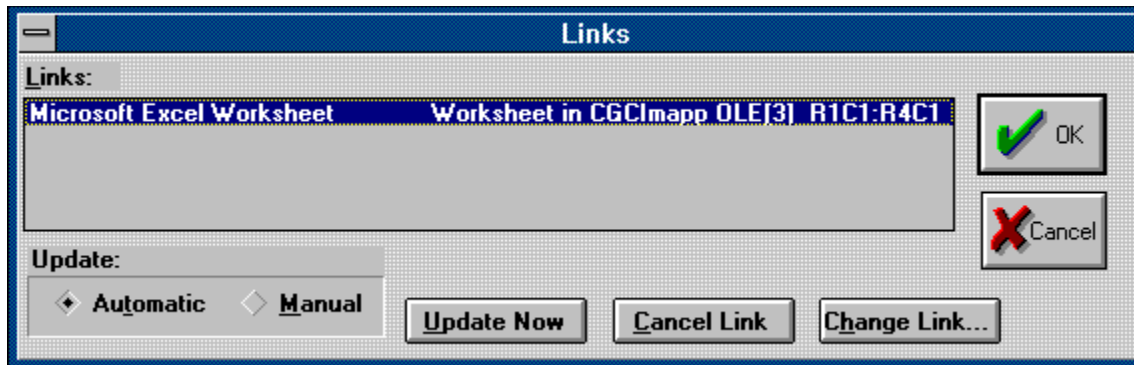
## **Edit Clear Menu Command**

**Edit Clear** menu command may delete any of the following:

- it will delete an OLE object if one is selected
- it will delete a portion of the image if a portion of the image is selected
- it will delete the entire image if nothing is selected.

## Edit Links Menu Command

**Edit Links** Dialog box displays a list of links to external documents stored in the active Image'n'Bits window. The embedded objects are not shown in the list.



### Update Options:

There are two Update option buttons: **Automatic** and **Manual**. These buttons reflect whether the currently selected links are automatically updated whenever the external linked file changes, or whether they must be manually updated.

An *automatic* link is updated automatically when the file it is linked to is changed in the OLE server or when the linked object is loaded and the OLE server is loaded with that object.

A *manual* link is a link that the user must explicitly update through this dialog (**Edit Links**) box. The update takes place when the link is selected and the user presses the **Update Link** button.

### Edit Link Dialog Buttons:

**Update Now** button updates all links selected in the Links list. In other words, the visual appearance of the OLE objects inside the Image'n'Bits document mirrors the current data in the linked server files.

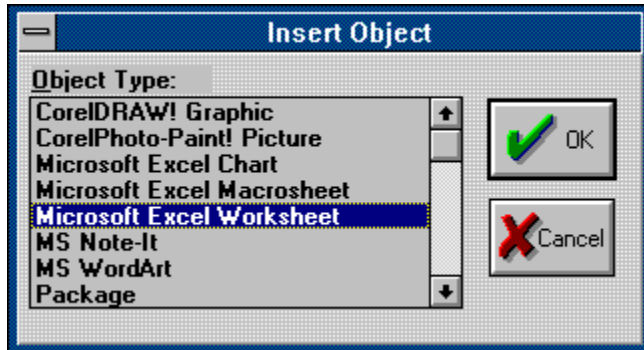
**Cancel Link** button permanently breaks the link between the Image'n'Bits document and the server. The linked object in the Image'n'Bits document is changed to a picture that can no longer be updated or edited using OLE related menu selections. This picture can still be edited using Cut, Copy, and Paste.

**Note:** There is no guarantee that the OLE object whose link was severed will continue to retain all of the data it contained when the link was intact.

**Change Link** button brings up a dialog box similar to the **Open** dialog box. The two screens are, in fact, identical except for the title. Change Link dialog box is used to repair links. For example, if a linked file is renamed or moved to a new location, this dialog let's the user reconnect the link in the IMA document, using the new name or the new location of the linked file. When the user chooses a file and presses OK in the **Change Link** dialog, Image'n'Bits disconnects the links that were selected in the **Links** list from their previous file and connects them to the newly chosen file.

## Edit Insert Object Menu Command

**Edit Insert Object** menu command presents the user with a list of OLE objects which can be inserted into the Image'n'Bits documents. After the user selects an object type and presses OK, Image'n'Bits inserts an empty OLE frame into the Image'n'Bits document and invokes the OLE server that is responsible for creating the selected object.





## Image Gamma Correction Menu Command

Gamma correction is a compensation for the monitor's non linearity. Perhaps this statement can be explained better by the following example. Let's say that all gray colors can be represented by numbers from 0 to 255 where 0 is black and 255 is white and all the gray scales are between those two colors. If the brush you were using was gray with a numerical value of 128 and you wanted that color to be just a little grayer you would probably intuitively make it 129 or 130. Well, computer monitors don't work in this "linear" fashion. Instead, the monitor's color values tend to jump in steps. The gamma correction factor takes "this is what I want" color value and returns "this is what you should use to get what you want" color value.

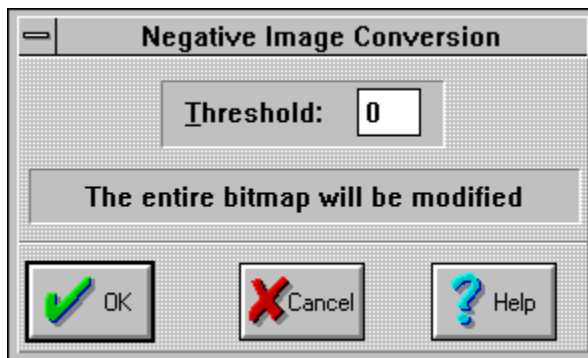
Gamma correction factors differ depending on the monitor and should be supplied by the manufacturer.

**Note:** Gamma Correction is applied to the entire image.

## Image Negative/Solarization Menu Command

Negative image operation, as the name implies, makes a negative of the image. You achieve it by setting the image threshold value to 0. Negative image is useful when image inversion might reveal an image detail that you could have missed during normal viewing. Applying negative operation to the same image twice will produce the original image.

Image solarization is very similar to the negative image operation, but the colors and visual effect is more interesting. Image solarization is achieved by setting the image threshold to a value other than 0 (value of 200 is a good start). By experimenting with the threshold value you can produce an image that is partially intact and partially solarized.



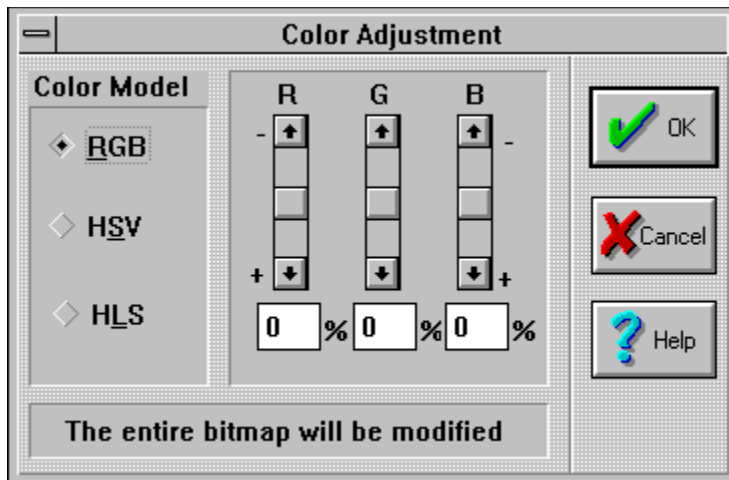
A maximum threshold value of 255 (highest intensity) will leave the image untouched.

**See Also**  
[Affected Image Area](#)

## **Image Area Affected by an Algorithm**

Image'n'Bits can operate on the entire image or just portions of it. An imaging algorithm dialog box (such as Negative Image) will indicate what image area will be affected. A particular algorithm will take into account the size and location of the clipping rectangle if a partial conversion is possible. In general, Image'n'Bits algorithms do not handle partial conversions on monochrome, 16 color, and 256 color images (and neither do other packages). Only true color (24 bits) images can be operated on piecemeal.

## Image Color Adjustment Menu Command



**Color Adjustment** dialog box allows you to add or reduce image color components using different color models. You have a choice of 3 color models: Red/Green/Blue (RGB), Hue/Saturation/Value (HSV), and Hue/Lightness/Saturation (HLS).

The RGB model is obvious. The components of the other two models are as follows: *Hue* distinguishes among colors such as red, green, purple. *Saturation* refers to how far the color is from a gray of equal intensity. For example, red is highly saturated; pink is relatively unsaturated. Royal blue is highly saturated; sky blue is relatively unsaturated. Pastel colors are relatively unsaturated. Basically, unsaturated colors include more white light than do the vivid, saturated colors. The last term that needs explanation is *Value*, which stands for brightness and is used in the HSV model.

### See Also

[Affected Image Area](#)

## **Image Line Art Conversion Menu Command**

**Line Art** menu command divides all colors within the image into two camps: lowest and highest intensity. The threshold value decides which colors go into which camp. For a gray scale image, line art operation produces black (lowest intensity) and white (highest intensity) images. For a color image, line art operation produces garish 8 color images.

### **See Also**

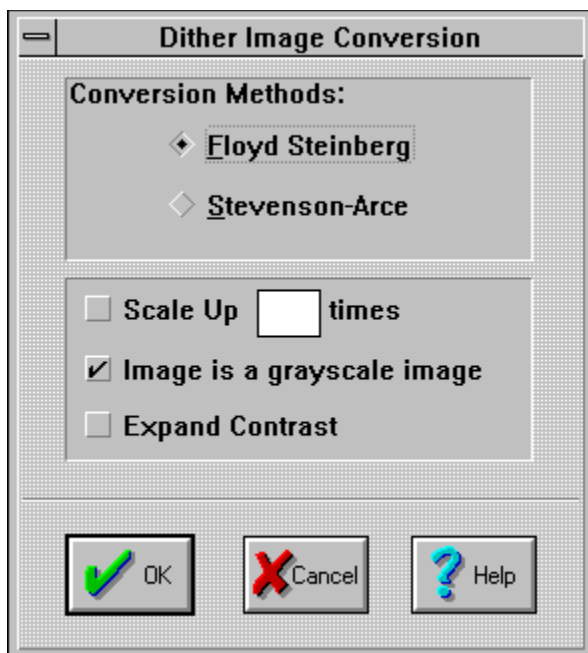
[Affected Image Area](#)

## Image Dither Menu Command

Dithering is a process applied to images in order to compensate for the color errors introduced when color content of the image exceeds that of a device. In other words, if the image you are displaying does not look like it is supposed to, dither it. You will be amazed at how good it can look with fewer colors. The above definition of dithering is a long way from being "classically" correct, but that is how it is used in Image'n'Bits and it does get the point across.

### Note:

- Dithering works on 24 bit (16 million colors) images only.
- Applying dithering makes sense on VGA (16 colors) systems only. Since the dithering algorithm works the same way regardless of the system's color capabilities, an image dithered on a 256 color system is not going to look any 'richer' in color than if it was dithered on a 16 color system.
- An image cannot be partially dithered. Therefore, the entire area of the image is going to be affected by this operation.



Image'n'Bits supports two dithering algorithms: **Floyd-Steinberg** and **Stevenson-Arce**. One algorithm is not better than the other - they are simply different and the dithering results are image specific. The Stevenson-Arce algorithm is inherently slower than the Floyd-Steinberg algorithm.

**Scale Up**, as the name implies, is a scaling factor. Applying a scaling factor when dithering produces an unexpected result: the new image looks as if it contains more detail than the original image. A scale factor of 2 (double image width and height) requires 4 times more memory than the original image. A scale factor of 3 (triple image width and height) requires 9 times more memory than the original image. Get the idea?

If the image you are dithering is grayscale, marking it as such using **Image is a grayscale image** button will speed up calculations. **Expand contrast** button slightly adjusts image contrast to make the image colors more pronounced.



## Image 16 Color Conversion Menu Command

When an image is converted from 'fewer colors' image format to 'more colors' image format, the color content of the image stays the same (if the color was not there to begin with, it cannot show up out of nowhere). Therefore you should not encounter any surprises when going from a monochrome (1 bit) image to a 16 color image (4 bits).

When an image is converted from 'more colors' format to 'fewer colors' format a color reduction operation is applied before the image's data can be reshuffled into a new format. A color reduction operation is a process where the existing set of colors is being approximated by a smaller set of colors -- there is plenty of room for errors here. To get the best color results, dither the image before converting it to a 16 color format image.



## Image 256 Color Conversion Menu Command

When an image is converted from 'fewer colors' image format to 'more colors' image format, the color content of the image stays the same (if the color was not there to begin with, it cannot show up out of nowhere). Therefore you should not encounter any surprises when going from a monochrome (1 bit) or a 16 color (4 bits) image to a 256 color image (8 bits). When an image is converted from 'more colors' format to 'fewer colors' format a color reduction operation is applied before the image's data can be reshuffled into a new format. A color reduction operation is a process where the existing set of colors is being approximated by a smaller set of colors -- there is plenty of room for errors here. To get the best color results, experiment with palettes from the **View** top level menu for a preview of the image before converting it to a 256 color format image.

**Note:** 24 bit to 256 color image conversion process will make use of the palette you select. By selecting the wrong palette it is possible to produce a 256 color image out of a 24 bit image which appeared to be grayscale.

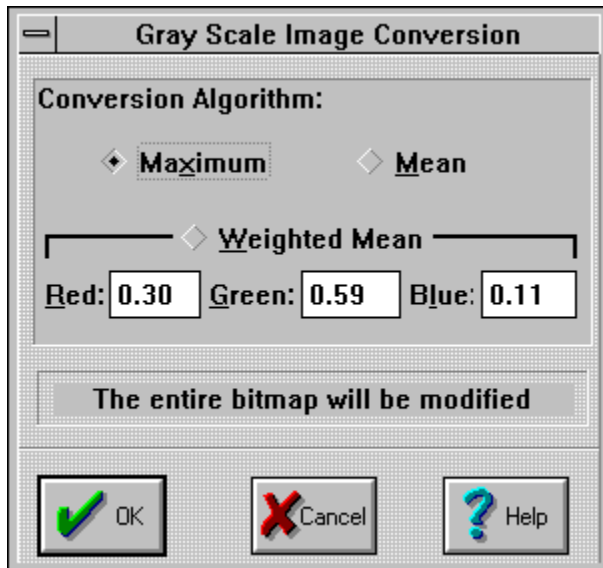
## **Image 24 bit Color Conversion Menu Command**

When an image is converted from 'fewer colors' image format to 'more colors' image format, the color content of the image stays the same (if the color was not there to begin with, it cannot show up out of nowhere). Therefore you should not encounter any surprises when going from a monochrome (1 bit), a 16 color (4 bits) image or a 256 color image (8 bits) to a 24 color image.

If the color content of the image stays the same, why bother converting it to a 24 bit image? The answer is that some operations apply to 24 bit images only. For example, you want to gray scale a portion of a 256 color image. Image'n'Bits (and other packages) can perform piecemeal operations on 24 bit images only. In this case, you would convert the source image to a 24 bit flavor and gray scale that new image.

## Image Gray Scale Menu Command

**Gray scale** dialog box converts color images to images containing shades of gray.



Image'n'Bits supports three methods of grayscaling: **maximum**, **mean**, and **weighted mean**. The **maximum** method tends to produce images a bit on the light side. The **mean** gray scaling method tends to produce images that are a bit on the dark side.

The **weighted mean** gray scaling method results depend on the percentage of each color channel (red, green, and blue) you assigned to be used in the gray scaling calculation. The percentage values of 0.30 for red, 0.59 for green, and 0.11 for blue are the recommended defaults. The sum of percentages should equal 1.

### See Also

[Affected Image Area](#)

## Edit Paste From File Menu Command

**Paste From File** menu command pastes (copies) a file from disk into your currently open document. It does not replace the existing image (as File Import command does), but adds to it.

**Paste From File** menu selection works differently depending on the lack or presence of the selection rectangle on the image you are pasting into. If the selection rectangle is defined on the image, the new clipping will be fitted into the selection rectangle stretching or shrinking the clipping as necessary. If a selection rectangle is not defined, the clipping will appear in the upper left corner of the image. In either case, the clipping can be dragged to its final destination by pressing the left mouse button on top of the clipping. To cancel the paste operation, press right mouse button (or press the escape key). To complete and accept the paste operation double click on the clipping using the left mouse button. While the paste operation is in progress all menu selections will be disabled including closing of the document window in which the image is displayed.

**Note:** Performing the paste operation on an image makes that image device dependent. For example, you are working on a 24 bit image on a 256 color system. You have copied one part of the image and pasted it on to another area of the image. If you check the image window's system menu you will see that the image you are working on has become a 256 color image. You shouldn't care since your system is only 256 colors anyway. On the other hand, you can apply a richer set of operations to a 24 bit image. Also, when printing to a color printer, your printout will contain only 256 colors.

### See Also

Affected Image Area

## View Menu Commands

### Palette Commands

[Uniform Distribution 884](#)

[Uniform Distribution 666](#)

[Popular](#)

[Median CutGray Scale](#)

[Stretch To Fit](#)

**Palette Menu Commands** applies to 24 bit images only. Palette menu command is a collection of 4 different algorithms used to extract a small subset of colors out of an image whose color content exceeds the video hardware capabilities. For example, let's say you have a 24 bit image that you are displaying on a 256 color Windows system. Which of the 256 colors (out of possible 16.7 million) should represent the bitmap? The palette menu commands solve that problem using different approaches.

## View Palettes Menu Commands

**Uniform Distribution 884** and **Uniform Distribution 666** menu commands are two algorithms that create a palette of colors uniformly distributed throughout the color spectrum regardless of the colors contained in the image. These two algorithms are the most inaccurate of the four algorithms included in this software release, but they are also the fastest.

**Gray Scale** palette menu selection creates a palette consisting of 256 shades of gray.

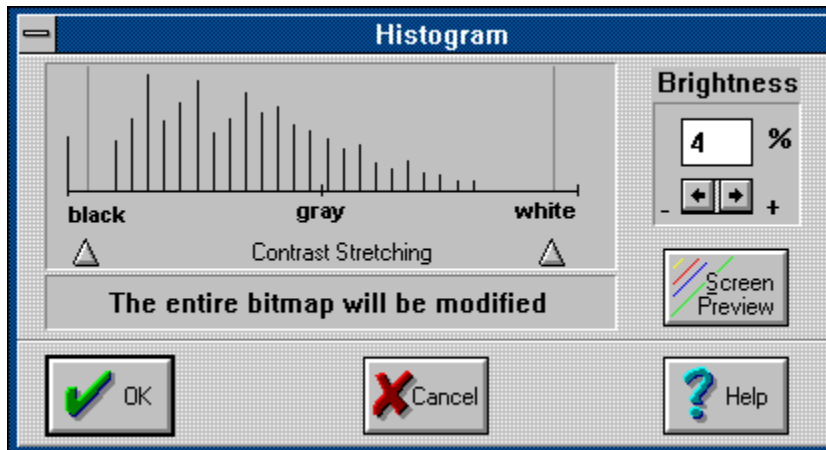
**Popular** and **Median Cut** algorithms derive their respective color palettes based on the color content of the image. They are more accurate than the first two, but they are also slower. One algorithm is not better than the other in terms of color accuracy. The result depends on a particular image.

After selecting a particular palette, the 24 bit image will be displayed using that palette.

**Note:** MS-Windows supports palettes only with 256 color device driver configuration, so don't expect to see anything different under the standard VGA 16 color configuration when selecting a different palette.

## Image Histogram Menu Command

**Histogram** Menu Command shows a histogram of the document's image. A histogram is a graph of the distribution of pixel-intensity values for an image or a portion of an image (simply put it is a count of each color occurrence). A histogram is a tool used to judge brightness and contrast of the image.



### How to read a histogram

If the histogram's graph lines are located mostly on the left, the image is too dark. If the histogram's lines are located mostly on the right, the image is too light. When the histogram's graph lines are occupying a narrow range, i.e. the histogram lines are bunched up together, the image has a low contrast. Look at different images and display their histograms to get a feeling for histograms. They can be a great tool in improving your images.

You can adjust image brightness and contrast right from the histogram dialog box (and preview the results). To adjust brightness, enter a percentage value into the brightness field. Positive percentages increase brightness, negative percentages decrease it. To apply contrast stretching, drag the contrast thumbtracks to the color/pixel intensity range you want to stretch. Press Screen Preview (or OK) button.

**Hint:** For best results, apply the brightening operation first, leaving contrast stretching thumbtracks in their initial positions of 0 and 255. After the image has been brightened (and the histogram has been changed to reflect this) apply contrast stretching. Notice how the image histogram changes after each of your adjustments.

Traditionally, histograms are applied to gray scale images only. If a histogram is to be performed on a color image, Image'n'Bits performs 'on the fly' gray scale conversion using the Maximum Gray Scale conversion method. If you want better control over the image (and a more accurate histogram) convert your image to gray scale yourself using the gray scale algorithm of your choice before working with the histogram dialog box.

**Note:** Image'n'Bits displays the histogram for the whole image. The clipping rectangle is ignored. The brightness and contrast adjustment operations are applied to the clipping area, if one is present, and to the whole image otherwise.

### See Also

Gray Scaling  
Affected Image Area

## **Stretch To Fit**

**Stretch To Fit** menu command fits (stretches or shrinks) the displayed image into the current window size. It is for visual effect only and does not do anything else. If you want to keep a bunch of images on the screen and yet keep them out of the way, **Stretch To Fit** is a convenient way of viewing the whole image in a small sized window.



## Options Menu Commands

**Options** menu commands contain menu selections which will help you to customize Image'n'Bits.

**Show/Hide status bar** menu selection will turn on or off the status bar at the bottom of the screen.

**Preferences** menu selection will bring up [Preferences Dialog Box](#).

## Preferences Dialog Box

**Apply gamma correction to all imported images.**

**Note:** gamma correction is not applied to the Clipboard (CLP) files.

### Metafiles

Metafiles come in two variations: Windows non-placeable and Aldus placeable.

Of the two formats, Aldus placeable metafile is a better and a more popular format since it is a Windows metafile with an extra header. The extra header contains metafile sizing information used to properly size and position the metafile file on the screen.

When importing a non-placeable metafile Image'n'Bits assumes a certain image size into which the incoming metafile will be "fitted". If the image looks too distorted, change the default image size and import the metafile again.

The Image'n'Bits default for exporting metafiles is Aldus placeable.

## Window Menu Commands

**Tile** menu command fits all Image'n'Bits windows side by side inside the main background window.

**Cascade** menu command arranges Image'n'Bits windows 'a la deck of cards', i.e. one behind the other.

**Arrange Icons** menu command neatly arranges iconized Image'n'Bits windows along the bottom of the main background window.

**Close All** menu selection is a short cut to closing all Image'n'Bits documents that are currently open. You will still be prompted if a document should be saved.

In addition to the above commands, Window Menu Popup may also contain titles of the currently open documents. Selecting a title will bring that window forward.

## **Embedded OLE Objects**

An embedded object is a "static" part of the compound document. It is a copy of the original and is not linked to the original in any way. It will remain unmodified even though the original object may have been changed. Double click on the object when you want to update or modify it.

### **See Also**

[Linked OLE Objects](#)

## **Linked OLE Objects**

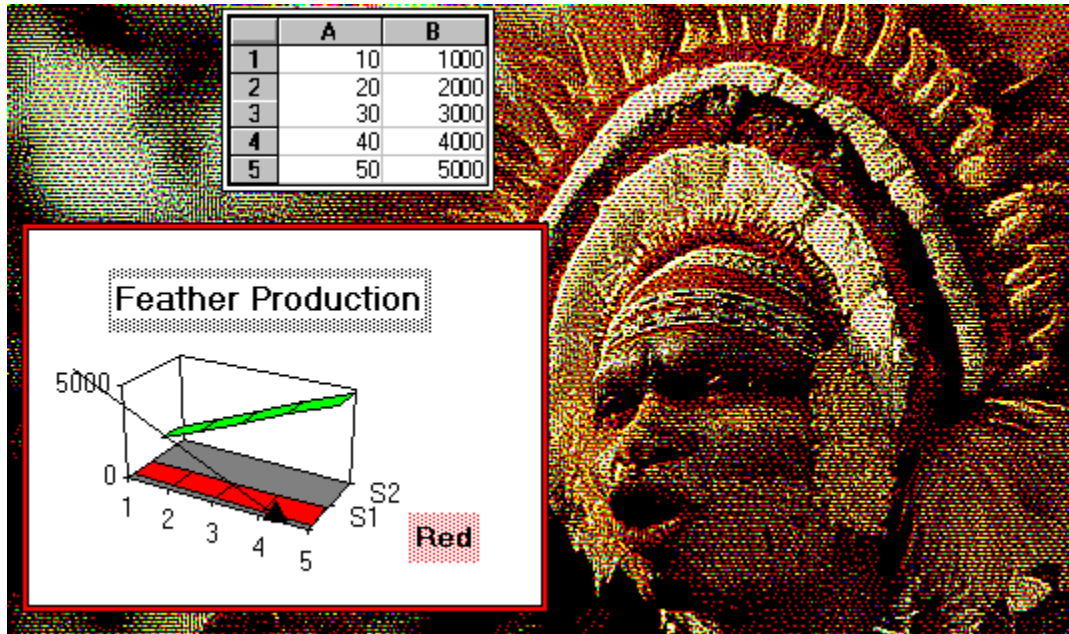
The compound document which contains linked objects does not physically contain those objects. Instead, it contains links to the applications that originally created those linked objects. The application that created the linked object is responsible for storing the linked object in its own file. When a compound document is opened all OLE linked objects within the document are updated.

### **See Also**

[Embedded OLE Objects](#)

## OLE Client

An OLE client is a program capable of incorporating other application's documents as part of its own documents. This approach allows the user to create compound application independent documents. The following sample document is an Image'n'Bits document which incorporates an Excel spreadsheet and an Excel chart.



## **Selection or Clipping Rectangle**

Selection rectangle is a rectangle which is created by clicking the left mouse button, holding it and dragging the mouse. A menu action, such as image gray scaling, affects the area of the selection rectangle if there is one, and the entire image if there is no selection rectangle defined.

## Drag and Drop

Image'n'Bits supports "drag and drop" capability. To open a file, you can drag it from the Windows File Manager and drop it on top of Image'n'Bits.

Image'n'Bits follows the following drag and drop rules:

- IMA files dropped on the Image'n'Bits application get opened in their own window.
- All other file formats (even those Image'n'Bits is capable of importing) are brought in as OLE objects and get attached to an already opened IMA document. In order to update and view the newly incorporated OLE object, double click on that object.



