Chapter 28
Importing and Exporting Graphics

Mathcad can read in graphic image files as data for you to manipulate mathematically and to view as pictures. You can process images as arrays in Mathcad and export arrays as image files to graphics editors or other applications. This chapter also describes how to create pictures in a Mathcad worksheet by reference to bitmap files, and how to import graphic images by copying them from another application and pasting them into Mathcad.

This chapter contains the following sections:

Reading and writing graphics files

Reading and writing grayscale BMP image files with *READBMP* and *WRITEBMP* and color BMP image files with *READRGB* and *WRITERGB*; viewing images in the picture operator. Additional image reading and writing functions in Mathcad Professional.

Creating pictures

Using the **Picture** command from the **Insert** menu to create a picture based on a matrix or bitmap file. Pasting in a graphic image from the clipboard.

Formatting pictures

Changing the positioning and size of existing pictures, and working with the palettes of 256-color bitmaps.

Reading and writing graphics files

Mathcad includes functions that allow you to read in a grayscale or color image as data. Once you have read an image into a Mathcad matrix, you can view it in Mathcad's picture operator or apply image processing techniques or other mathematical manipulations. And once you finish processing a matrix, you may save the data to disk as an image for later use in Mathcad or in graphics applications.

While the functions described in this section are designed specifically to read and write image files, they share many of the general behaviors of the functions for reading and writing ASCII data described in Chapter 19, "Data Management."

Reading, viewing, and writing BMP files

To import a bitmap (BMP format) file as data into Mathcad, use one of these functions:

Function	Meaning
READBMP(*file*)	Create a matrix containing a grayscale representation of the bitmap image in *file*. Each element in the matrix corresponds to a pixel. The value of a matrix element determines the shade of gray associated with the corresponding pixel. Each element is an integer between 0 (black) and 255 (white).
READRGB(*file*)	Create a matrix in which the color information in *file* is represented by the appropriate values of red, green, and blue. This matrix consists of three submatrices, each with the same number of columns and rows. Three matrix elements, rather than one, correspond to each pixel. Each element is an integer between 0 and 255. The three corresponding elements, when taken together, establish the color of the pixel.

You must use all uppercase letters to type these function names; alternatively, choose **Function** from the **Insert** menu and double-click on the function name in the scrolling list.

The argument you supply to *READBMP* or *READRGB* is a string expression—or a variable to which a string is assigned—as described in "Arguments to file access functions" on page 454. The string can correspond to either:

- the name of a bitmap (BMP format) file in the working directory of the Mathcad worksheet you're currently working on; or
- a full or relative path to a bitmap file located elsewhere on a local or network file system.

Each function returns a matrix of numbers used to represent the image.

For example, to create a matrix **M** of numbers corresponding to the image in the color image file **`data2.bmp`**, type either

`M:=READBMP("data2.bmp")` or

`M:=READRGB("data2.bmp")`

In the first case **M** will be a grayscale representation of the image; in the second, **M** will consist of three submatrices corresponding to the red, green, and blue components of the color image.

As a second example, to read the grayscale image **`blur.bmp`** in the **`C:\DATA`** directory into the matrix **B**, pass the following string expression to *READBMP*:

`B:=READBMP("C:\DATA\blur.bmp")`

See Chapter 8, “Variables and Constants,” for more information about creating string expressions.

To divide the matrix for a color image into its red, green, and blue components, use the formulas shown in Figure 28-1 that use the *submatrix* function. In this example the color bitmap file **`monalisa.bmp`** is read into a grayscale matrix **gray** as well as the packed RGB matrix **packed** and then converted into three submatrices called **red**, **green**, and **blue**.

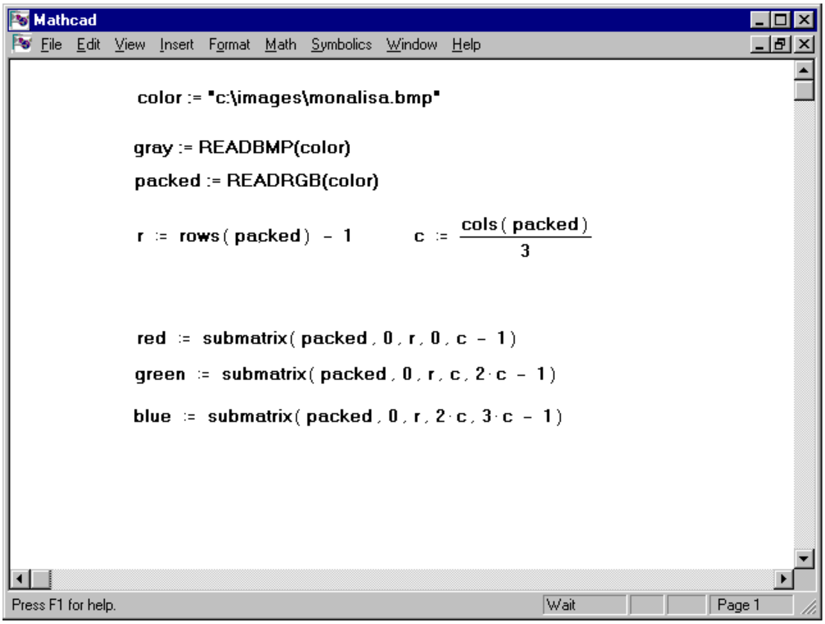
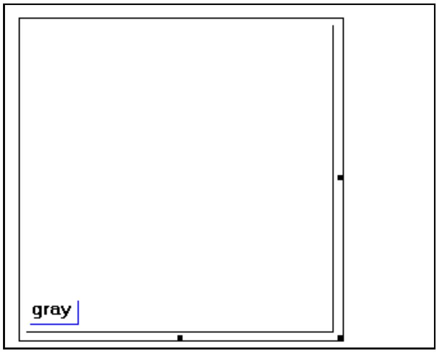


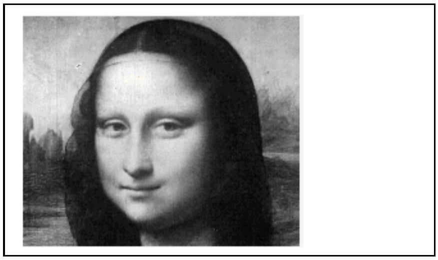
Figure 28-1: Reading in a color image as both grayscale and packed RGB matrices.

Once you have read an image file into Mathcad, you can use the *picture operator* to view it. First click in a blank part of your worksheet and then:

- Choose **Picture** from the **Insert** menu.
- Type the name of the matrix in the placeholder at the bottom left of the picture operator.



- Click outside the picture region to see the image.



To display an image in color, you must provide the picture operator with the names of three matrices containing the red, green and blue color values of the image. Otherwise, the displayed image will be in grayscale. For example, to display the image in Figure 28-1, you would choose **Picture** from the **Insert** menu and type **`red,green,blue`** into the placeholder:

If you want to display only the red components of the image that was used for Figure 28-1, you still must include the green and blue submatrices, but they both should contain only zeros.

Keep in mind that the colors of the image that you ultimately see may be distorted to the extent that you don't display 256 colors on your monitor.

If you have been working with submatrices for a color file, you can recombine them by defining an *augment3*(**X**, **Y**, **Z**) function:

`augment3(X, Y, Z) := augment(X, augment(Y, Z))`

X, **Y**, and **Z** are the names of the submatrices and should have the same number of rows and columns. The resulting matrix will then have the same number of rows as **X** but three times the number of columns.

Once you finish processing a matrix, you may want to save it to disk for later use. To do this, use one of the following functions:

Function	Meaning
WRITEBMP(*file*)	Create a grayscale BMP file from the matrix.
WRITERGB(*file*)	Create a color BMP file from a matrix in which the image is stored in RGB format.

You can use Mathcad's *augment* function (or the *augment3* function defined above) to combine submatrices at the time you are saving a file to your disk. For example, the following saves the submatrices that were created in Figure 28-1 as one 24-bit color image file called **`nucolor.bmp`**:

`WRITERGB("nucolor.bmp"):=augment(red,augment(green,blue))`

WRITEBMP and *WRITERGB* each take a string expression as described in "Arguments to file access functions" on page 454. The string can correspond to either:

- the name of a bitmap (BMP format) file in the working directory of the Mathcad worksheet you're currently working on; or
- a full or relative path to a bitmap file located elsewhere on a local or network file system.

Additional image reading and writing functions

Pro

In addition to the functions for reading and writing bitmap files described above, Mathcad Professional includes an assortment of more specialized functions for reading images or image components, including functions for reading images in GIF, JPG, and TGA formats. Your choice of which function to use will depend on:

- Whether you want to create a Mathcad matrix in which the image is separated into: red, green, and blue (RGB); hue, lightness, and saturation (HLS); or hue, saturation, and value (HSV) components.
- Whether you want to read in the entire image or just one component (for example, only the red values) of the image.

The functions used to read images work in exactly the same way as *READBMP* described in the previous section, but each recognizes images in the following formats: BMP, GIF, JPG, and TGA. Each function a string expression as described in "Arguments to file access functions" on page 454. The string can correspond to either:

- the name of an image (BMP, GIF, JPG, or TGA format) file in the working directory of the Mathcad worksheet you're currently working on; or
- a full or relative path to an image file located elsewhere on a local or network file system.

Each function returns a matrix of numbers used to represent the image.

	Function	Meaning
Pro	READ_IMAGE(*file*)	Creates a matrix containing a grayscale representation of the image in *file*. To read an image in color, use one of the functions below.
Pro	READ_HLS(*file*) READ_HSV(*file*)	Creates a matrix in which the color information in *file* is represented by the appropriate values of hue, lightness, and saturation (HLS) or hue, saturation, and value (HSV).
Pro	READ_RED(*file*) READ_GREEN(*file*) READ_BLUE(*file*)	Extracts only the red, green, or blue component from a color image. The result has one-third the number of columns that the matrix returned by *READ_RGB* would have had.
Pro	READ_HLS_HUE(*file*) READ_HLS_LIGHT(*file*) READ_HLS_SAT(*file*)	Extracts only the hue, lightness, or saturation component from a color image. The result has one-third the number of columns that the matrix returned by *READ_HLS* would have had.
Pro	READ_HSV_HUE(*file*) READ_HSV_SAT(*file*) READ_HSV_VALUE(*file*)	Extracts only the hue, saturation, or value component from a color image. The result has one-third the number of columns that the matrix returned by *READ_HSV* would have had.

In addition to the *WRITEBMP* and *WRITERGB* functions described in the previous section, Mathcad Professional has functions for creating color BMP files out of matrices in which the image is stored in HLS or HSV format.

	Function	Meaning
Pro	WRITE_HLS(*file*)	Create a color BMP file out of a matrix in which the image is stored in HLS format.
Pro	WRITE_HSV(*file*)	Create a color BMP file out of a matrix in which the image is stored in HSV format.

Like *WRITEBMP* and *WRITERGB*, these functions each take a string expression as described in “Arguments to file access functions” on page 454. The string can correspond to either:

- the name of a bitmap (BMP format) file in the working directory of the Mathcad worksheet you’re currently working on; or
- a full or relative path to a bitmap file located elsewhere on a local or network file system.

To use these functions, you write an assignment expression with the left-hand side containing the call to the *WRITE_HLS* or *WRITE_HSV* function and the right-hand side containing a matrix expression. For example, to create the 24-bit color image file **`process.bmp`** in the current directory from matrices **H**, **L**, and **S** corresponding to the hue, light, and saturation components of the image, enter:

```
WRITE_HLS("process.bmp"):=augment(H,augment(L,S))
```

Creating pictures

You can create a picture in a Mathcad worksheet in the following ways:

- By using the picture operator and supplying it either the name of a Mathcad matrix or the name of an external bitmap file as a string expression, or
- By pasting in a graphic image from another application via the clipboard.

Creating a picture from a matrix

You can view any matrix in Mathcad as a picture by using the picture operator:

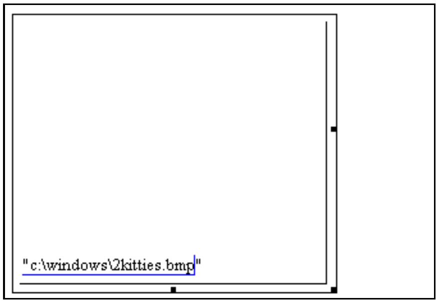
- Click in a blank space in your Mathcad worksheet.
- Choose **Picture** from the **Insert** menu.
- Type the name of a matrix in the placeholder at the bottom of the operator.

This is most useful when you import graphics files into Mathcad as matrices as described in “Reading and writing graphics files” on page 586. For example, you can use the *READBMP* function to read a bitmap file into a matrix, and then use the picture operator to see the picture in Mathcad. See the figure on page 588 for an example.

Creating a picture by reference to a bitmap file

To create a picture directly from a bitmap file without reading it first into a matrix, click in a blank space in your worksheet and then:

- Choose **Picture** from the **Insert** menu to insert the picture operator.
- In the placeholder, type a string expression containing the name of a bitmap file in the current directory, or type a full path to a bitmap file.



- Click outside the picture operator and the bitmap will appear in your worksheet.

Each time you open the worksheet or calculate the worksheet, the bitmap file will be read into the picture operator. If you modify the source bitmap file, you must recalculate your worksheet to see the modified image. If you move the source bitmap file, Mathcad will not be able to display the picture.

Importing graphic images from the clipboard

You can copy any image from another application to the clipboard and paste it into Mathcad in one of the formats put on the clipboard at the time of copying. If you use the **Paste** command on Mathcad's **Edit** menu to paste in an image from the clipboard (or use drag-and-drop from another application), you will often paste a linked *OLE object* into your Mathcad worksheet, as discussed in Chapter 4, "Worksheet Management." When you double-click on a linked OLE object, you activate the application that created the object and are able to edit the object in your Mathcad worksheet.

This section describes using the **Paste Special** command on the **Edit** menu to paste graphic images into Mathcad worksheets in noneditable formats: as pictures (metafiles) or bitmaps. A metafile, which is strictly a Windows graphic format, can be resized in Mathcad without undue loss of resolution, whereas a bitmap is usually viewed best only at its original size. A device-independent bitmap, or DIB, is stored in a bitmap format that is portable to other operating systems.

To paste a graphics image from the clipboard into Mathcad, do the following:

- Place the graphics image on the clipboard, usually via a **Copy** command on the **Edit** menu. Many Windows applications have this feature.
- Click the mouse wherever you want the image in your Mathcad worksheet.
- Choose **Paste Special** from the **Edit** menu, and choose one of the available formats. If you do not wish to paste a linked OLE object, choose "Picture (metafile)" or "Device Independent Bitmap" from the list in the Paste Special dialog box.
- Click "OK." Mathcad creates a picture region and puts into it the image stored on the clipboard.

The format choices in the Paste Special dialog box will vary, depending on the application from which you originally copied a selection.

Mathcad stores the color depth—the number of colors in the image—at the time you paste it into a worksheet. This means that you can safely resave any worksheets that contain color images on systems that have different color displays, either fewer or more colors. The images will continue to display at the proper color depth on the systems that created the worksheets.

When you import directly from the clipboard, the picture information is stored as part of the Mathcad worksheet. This makes the document take up more disk space. It also means that when you copy the worksheet, the picture information travels along with it.

Formatting pictures

As discussed in previous sections of this chapter, you create pictures in a Mathcad worksheet from matrices, bitmap files, or graphic images pasted from the clipboard. This section describes your options for formatting a picture once you've created it.

Resizing pictures

To resize a picture region, do the following:

- Click the mouse inside the picture region to select it.
- Move the mouse pointer to one of the handles along the edge of region. The pointer will change to a double-headed arrow.
- Press and hold down the left mouse button. With the button still held, drag the mouse in the direction you want the picture region to be stretched.

When you change the size of the picture region, the picture inside may be distorted. If you resize the picture by dragging diagonally on the handle in the lower right corner, you will preserve the aspect ratio—the ratio of height to width—of the original picture.

You can also avoid distortion by restoring the original size of the picture region:

- Double-click on the picture itself, or choose **Properties** from the **Format** menu. This brings up the Properties dialog box.
- Click on the box labeled "Display at Original Size."
- Click the "OK" button. Mathcad changes the size of the picture to match that of the original image.

Framing a picture

Mathcad allows you to place a border all the way around a picture region. To do so:

- Double-click on the picture itself, or choose **Properties** from the **Format** menu. This brings up the Properties dialog box.
- Click on the box labeled "Show Border."
- Click the "OK" button. Mathcad draws a border around the picture region.

Deleting and moving pictures

Use the **Edit** menu commands to cut, delete, or copy a picture region.

To cut or delete a picture region:

- Click inside the region to select it. This will place a selection box around the picture.
- Press [`Ctrl`]**X** or choose **Cut** from the **Edit** menu. Choose **Delete** from the **Edit** menu to leave the clipboard unaffected.

To copy a picture region:

- Click inside the region to select it. This will place a selection box around the picture.
- Press [**`Ctrl`**]**`C`** or choose **Copy** from the **Edit** menu.
- Click the mouse wherever you want to place a copy of the region and press [**`Ctrl`**]**`V`** or choose **Paste** from the **Edit** menu.

To move a picture region:

- Click inside the region to select it. This will place a selection box around the region.
- Place the pointer on the border of the box. The pointer will turn into a small hand.
- Press and hold down the mouse button.
- Without letting go of the button, move the mouse to where you'd like the picture region.

Note that unlike some applications, Mathcad treats the picture region and the image within it as a single region, not as separate frames. Therefore, when you press [**`Ctrl`**]**`C`** or [**`Ctrl`**]**`X`**, Mathcad places both the image and the picture region into the clipboard.

Controlling color palettes

If you are using a 256-color display and have color bitmaps in your Mathcad worksheets, Mathcad by default uses a single 256-color palette to display all the bitmaps in your worksheets. This is the same default color palette Mathcad uses for displaying the rest of the Mathcad screen, and is suitable for most pictures.

This default color palette, however, may not be the exact one that any color bitmaps in a worksheet were designed to use. To improve the appearance of bitmaps in your worksheet, you can tell Mathcad to optimize its default color palette so that it chooses the best possible 256 colors to display bitmaps in the worksheet. To do so:

- Choose **Color ⇒Optimize Palette** from the **Format** menu. Now Mathcad will survey the pictures in the worksheet and generate an optimal 256-color palette to use for all of them.
- Make sure that **Color ⇒Use Default Palette** in the **Format** menu has a check mark. Then Mathcad will use the new default palette it generates.

If you don't want Mathcad to use one color palette for all color bitmaps, choose **Color ⇒Use Default Palette** in the **Format** menu. The check mark next to this option will be removed, and Mathcad will use whatever color palette is best for each picture. The disadvantage to unchecking **Use Default Palette** is that the screen may flash as you scroll through a Mathcad worksheet and each 256-color bitmap loads a new palette. We therefore recommend that you keep a check mark next to the **Color ⇒Use Default Palette** option in the **Format** menu if you are working in 256-color mode.

If your display driver supports a greater number of colors, the palette setting options on the **Format** menu are grayed out.