

IMS-2D Image Simulation

The 2D Image Simulation (IMS) option simulates the appearance of a 2D input object as it is imaged through the optical system defined in CODE V. For example, if you have a fisheye lens defined in CODE V, you can define a 2D object for input into the IMS feature, and the program will show you how that object will be imaged by the fisheye lens, including distortion and other effects.

When to Use the 2D Image Simulation Option

With the 2D Image Simulation (IMS) option, you can:

- View the appearance of a 2D object as it is imaged by the optical system defined in CODE V, including the following types of effects: diffraction and aberration blurring, image distortion, keystone distortion, longitudinal and lateral color, relative illumination variations, and blurring due to detector dimensions.
- Determine the image orientation of an optical system with multiple folds.
- Specify an input object that can be sized to cover the full field, or a sub-area of the field. You can position, orient and scale the input object according to your needs. Several pre-defined input objects are provided with CODE V.
- Generate a full-field PSF diagnostic grid, which shows the relative variation of PSFs across the defined image area, the correct relative position of the PSFs to illustrate distortion, and the PSF intensities, which are normalized across the field of view so that they are easily visible. The PSFs can also be convolved with a rectangular detector to include the effects of detector blurring.
- Display the output image in either gray scale or RGB formats, and save the output image in various standard graphics file formats.

Default Operation

For the resident lens system, the grayscale image of a U.S. Air Force resolution target is generated. By default, the original target is also displayed. The semi-diagonal of the object file is mapped to the maximum field of view for the lens.

Command Mnemonics (alphabetical)

AZI	BWL	BWT	CME	DEX	DEY
FCO	GRI	GWL	GWT	IDP	NRD
OBH	OBJ	ODP	PDP	PMX	PMY
RWL	RWT	SVI	SYM	TGR	XOF
YOF					

IMS Procedure Overview

The procedure for running an image simulation is:

1. Define the input object. This includes specifying the bitmap file (.BMP format) to be used as the 2D object, and defining the field size and position of the file with respect to one of the four field coordinate systems supported by CODE V.
2. Define the Point Spread Function (PSF) controls. This includes defining standard PSF controls such as the FFT grid size, number of rays across the diameter, and the focal plane increment. You can also define the grid of PSFs to be computed over the object, and optionally enter the detector size to include detector blurring effects.
3. Define the color computational method. This includes polychromatic intensity, RGB, or three-wavelength color. For example, polychromatic intensity performs a polychromatic PSF for all system wavelengths. The result is an 8-bit gray scale image. This is the default.
4. Define the output. By default, the object file and image simulation result will be displayed. You can also choose to display the PSF map, and save the image simulation result into an output file in bitmap (BMP), JPEG, and Portable Network Graphics (PNG) formats to an 8-bit depth for monochrome or 24-bit depth for color. In addition, you can optionally save the image as a PNG data file that includes header information for post-processing. The data file is saved to a 16-bit depth for monochrome or 48-bit depth for color.

Launching IMS from the CODE V Toolbar

To launch the 2D Image Simulation (IMS) option, you can click the IMS button on the Options toolbar:



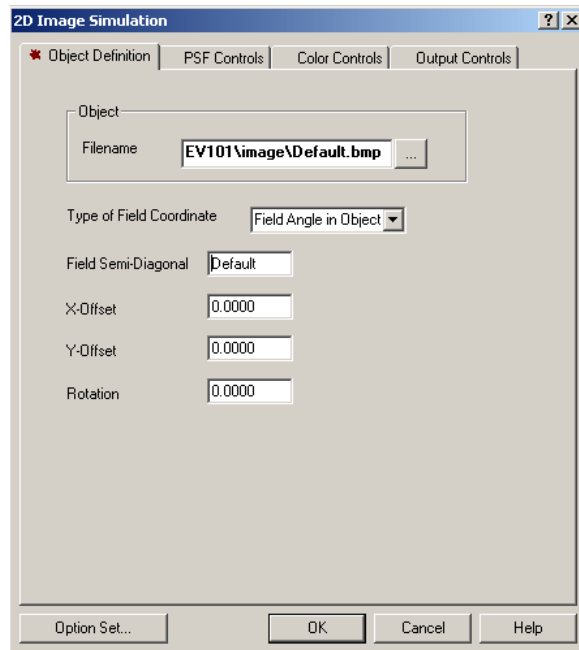
To add this entire toolbar to your default toolbar configuration, or just add the IMS toolbar button, choose the **Tools > Customize** menu, and click the **Toolbars** tab in the **Customize** dialog box. For details about how to add toolbars or toolbar buttons to your workspace, click the **Help** button in the **Toolbars** tab.

Defining the Input Object

To define the input object for the 2D Image Simulation (IMS) option, choose the **Analysis > Diffraction > 2D Image Simulation** menu. The **2D Image Simulation** dialog box displays, with the **Object Definition** tab automatically selected. Define the controls on this tab as described in the following table.



Note: When defining the input object file, it is important to understand issues related to the gamma associated with the file and display. See “Gamma Issues for Displaying, Printing, and Saving Files” on page 19-245 for details.

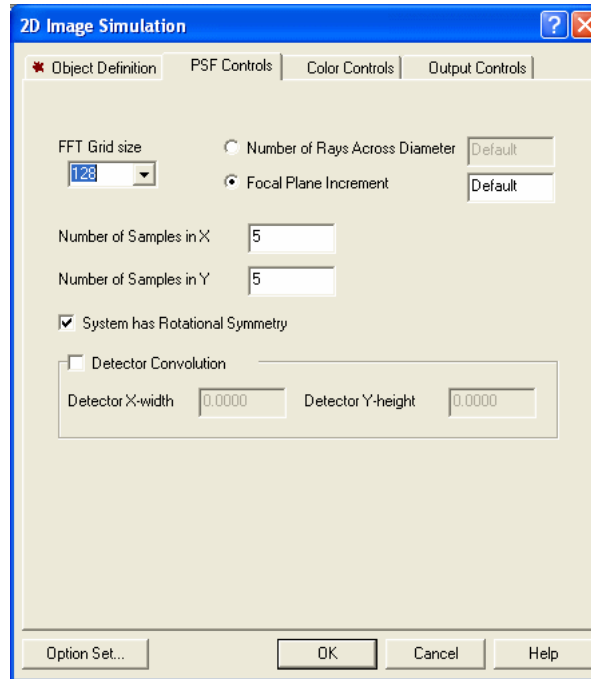


Command Syntax		
Screen Control	Explanation	Default
OBJ filename		
Object Filename	Defines the 2D object or “scene” file to use as input. Several example input object files are provided with CODE V in the image subdirectory of the CODE V main program directory: C:\CODEV102\image where C:\CODEV102 is the CODE V installation directory. You save your own bitmap image as <i>default.bmp</i> in the image subdirectory, and IMS will automatically load this file as your default input object.	U.S. Air Force 1951 resolution target (default.bmp)
FCO OBJ IMG RIH ANG		
Type of Field Coordinate	Selects the field coordinate definition used by the OBH, YOF, and XOF commands: • Object Height (OBJ) • Paraxial Image Height (IMG) • Real Image Height (RIH) • Field Angle in Object Space (ANG)	Field specification type defined in the LDM (SPC command)

Command Syntax		
Screen Control	Explanation	Default
OBH object_ht....z		
Field Semi-Diagonal	Maps the half-diagonal of the input object file to the specified field size, as defined by the field coordinates (FCO) command. The object is assumed to be centered at the XOF and YOF field designation, and the diagonal extends from -OBH to +OBH. If OBH is omitted, then the object file half-diagonal is scaled to the maximum half-field in object space. Note: Default apertures will be based on the LDM field specifications, and not the field specifications implied by the OBH command.	Maximum half-field specification
XOF object_offset_x....z		
X-Offset	Defines the offset of the center of the object in the X-direction. This command can be used to center the digital object file (i.e., the object file that represent the scene, or a portion of the scene) to a different position in the field of view of the lens. The units used by XOF are defined by the FCO command.	0
YOF object_offset_y....z		
Y-Offset	Defines the offset of the center of the object in the Y-direction. This command can be used to center the digital object file (i.e., the object file that represents the scene, or a portion of the scene) to a different position in the field of view of the lens. The units used by YOF are defined by the FCO command.	0
AZI object_orientation_degrees....z		
Rotation	Rotates the orientation of the digital object data. The rotation is performed on the digital data, in a 2D planar coordinate system (i.e., an XY, or gamma rotation following standard CODE V gamma sign conventions). For curved or tilted object surfaces, this rotation is done prior to mapping the object onto the curved or tilted surface.	0

Defining Point Spread Function (PSF) Controls

To display a full-field PSF diagnostic grid along with the image simulation, choose the **Analysis > Diffraction > 2D Image Simulation** menu. The **2D Image Simulation** dialog box displays. Click the **PSF Controls** tab, and define the controls as described in the following table.



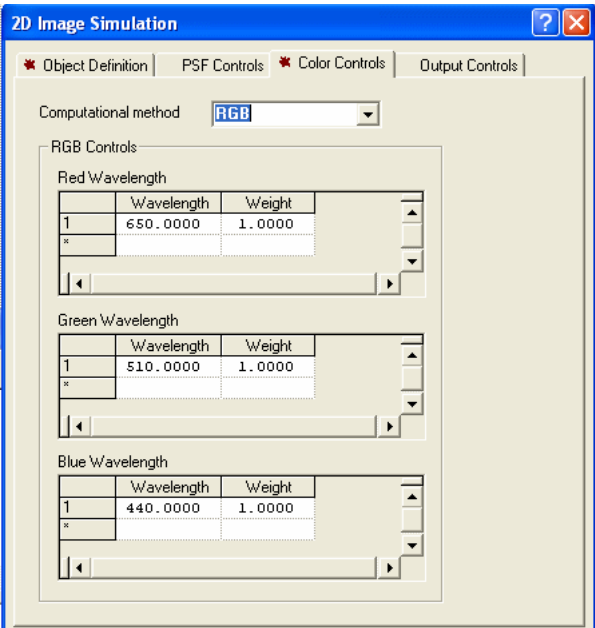
Command Syntax		
Screen Control	Explanation	Default
TGR num_fft_grid_pow_of_two....z		
FFT grid size	Specifies transform grid size (TGR). Range: 16 or more, must be a power of 2. Unless overridden by Number of rays across diameter (NRD) or Focal plane grid increment (GRI), also determines the number of rays traced in proportion to $TGR^{**}2$. Use to improve accuracy for systems with large or unusual obscurations, or to increase the size of the image area analyzed relative to the Airy disc. The grid size is independent of the number of samples in the input object file.	128

Command Syntax		
Screen Control	Explanation	Default
NRD num_rays_across_diameter....z		
Number of rays across diameter	Specifies number of rays traced across the pupil diameter for the shortest wavelength; other wavelengths are proportionately smaller to make image plane grids match. See also Focal plane grid increment (GRI) description. Range: Recommended - no more than 2 times up or down from the default; look out for artifacts of extreme choices. Use either Number of rays across diameter (NRD) or Focal plane grid increment (GRI), if desired, but not both.	TGR/2 at short wavelength.
GRI image_grid_spacing....z		
Focal plane increment	Specifies grid spacing measured on the image surface - is an alternative to Number of rays across diameter (NRD). Use this command to choose a more convenient number than the default. Range: Recommended - no more than 2 times up or down from the default; look out for artifacts of extreme choices. Use either Number of rays across diameter (NRD) or Focal plane grid increment (GRI), if desired, but not both.	Shortest wavelength * fnumber/2. (Airy disc diameter for shortest wavelength spans 4.88 grid increments regardless of TGR value.)
PMX num_samples_x		
Number of Samples in X	Defines the number of samples in the X-direction in the field at which the PSF is computed. The PSF varies with position in the image, and this command allows you to control how many different sample PSFs are computed and used in the image simulation. On the command line, if only PMX is entered, then PMY is assumed to take the same value. For rotationally symmetric systems (SYM ROT), the maximum of PMX and PMY is used to determine the radial number of samples along a radius from the center to the edge of the field.	5
PMY num_samples_y		
Number of Samples in Y	Defines the number of samples in the Y-direction in the field at which the PSF is computed. The PSF varies with position in the image, and this command allows you to control how many different sample PSFs are computed and used in the image simulation. On the command line, if only PMY is entered, then PMX is assumed to take the same value. For rotationally symmetric systems (SYM ROT), the maximum of PMX and PMY is used to determine the radial number of samples along a radius from the center to the edge of the field.	5

Command Syntax		
Screen Control	Explanation	Default
SYM ROT NON		
System Has Rotational Symmetry	Select to define the system as having rotational symmetry (regardless of actual symmetry). By default, the symmetry state is determined from the lens data. This setting specifies how IMS will process the PSF array. For rotationally symmetric systems, produces a swept array of PSFs; this setting is only valid when the x- and y-offset values (XOF and YOF) are equal to zero. For non-rotationally symmetric systems, produces a grid of PSFs based on the number of samples in X and Y (PMX and PMY). From the command line, use either the ROT or NON keywords: ROT – System has rotational symmetry. NON – System has no symmetry. Since IMS performs a more efficient calculation if the system has rotational symmetry, SYM ROT is recommended.	Calculated from lens.
DEX detector_size_X		
Detector Convolution: Detector X-width	In the GUI, select the Detector Convolution checkbox to include the effect of detector blurring in the PSF calculation. Specifies the X-dimension of the rectangular detector to be convolved with the system PSF. If you only enter the X-width (DEX), IMS will assume the same value for the Y-width (DEY), and the detector will be square. Range: Less than 1/2 the full grid X dimension.	0.0
DEY detector_size_Y		
Detector Convolution: Detector Y-width	In the GUI, select the Detector Convolution checkbox to include the effect of detector blurring in the PSF diagnostic grid. Specifies the Y-dimension of the rectangular detector to be convolved with the system PSF. If you only enter the Y-width (DEY), IMS will assume the same value for the X-width (DEX), and the detector will be square. Range: Less than 1/2 the full grid Y dimension.	0.0

Defining Color Controls

To define the color computational method for the 2D Image Simulation (IMS) option, choose the **Analysis > Diffraction > 2D Image Simulation** menu. The **2D Image Simulation** dialog box displays. Click the **Color Controls** tab, and define the controls as described in the following table.



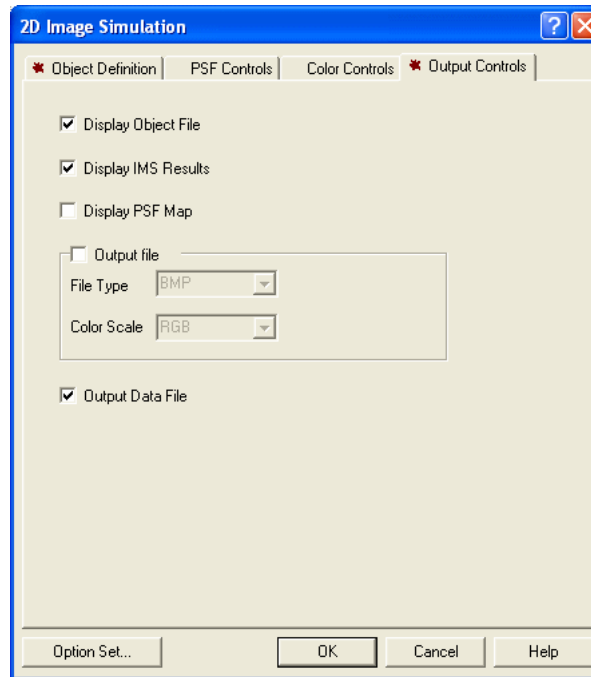
Command Syntax		
Screen Control	Explanation	Default
CME PIS RGB COL		
Computational method	Specifies the method used to compute the output image. Polychromatic Intensity (PIS) Computes a single PSF that is a weighted combination of the monochromatic PSFs for each wavelength specified in the lens system data. The corresponding wavelength weights are used to weight these component PSFs. The result is a single-band intensity image (i.e., a grayscale, 8-bit, intensity image). RGB Computes a separate PSF for the red, green, and blue contributions to the image. These component PSFs are based on the wavelengths specified in the color model. The simplest case uses a single wavelength for red, a single wavelength for green, and a single wavelength for blue with unity weighting. The result is a 24-bit color image. 3-Wavelength Color (COL) Computes a separate PSF for the red, green, and blue contributions to the image. The three PSFs are those computed for long, reference, and short LDM wavelengths. The result is a 24-bit color image.	Polychromatic Intensity (CME PIS)

Command Syntax		
Screen Control	Explanation	Default
RWL red_wavelength_nm....w		
RGB Controls: Red Wavelength	Specifies one or more wavelengths that are to be used in computing the red PSF for the RGB simulation method. You can enter up to 20 red wavelength values (RWL commands). The red PSF is the linear combination of the PSFs at each of the wavelengths specified with the RWL command, appropriately weighted according to the wavelength weight. From the command line, the command syntax element ...w means that you can enter several wavelengths at once (RWL 650 546 450), or you can include the Wi Wi.j wavelength qualifier prior to the value (RWL W1 450). Note: The number of wavelengths is limited to 20 for red, 20 for green, and 20 for blue, allowing a total of 60 wavelengths to define the three spectral bands.	650 nm for CME RGB. Not used for Polychromatic Intensity (CME PIS) or 3-Wavelength Color (CME COL).
RWT red_wavelength_wt....w		
RGB Controls: Red Wavelength Weight	Associates a weight with the PSF computation at the specified wavelength. The red PSF used in the RGB simulation will be the sum of the PSFs computed for each of the wavelengths specified by RWL commands with the appropriate weights, specified by the RWT command.	1.0
GWL green_wavelength_nm....w		
RGB Controls: Green Wavelength	Specifies one or more wavelengths that are to be used in computing the green PSF for the RGB simulation method. You can enter up to 20 green wavelength values (GWL commands). The green PSF is the linear combination of the PSFs at each of the wavelengths specified with the GWL command, appropriately weighted according to the wavelength weight. From the command line, the command syntax element ...w means that you can enter several wavelengths at once (GWL 650 546 450), or you can include the Wi Wi.j wavelength qualifier prior to the value (GWL W1 450). Note: The number of wavelengths is limited to 20 for red, 20 for green, and 20 for blue, allowing a total of 60 wavelengths to define the three spectral bands.	510 nm for CME RGB. Not used for Polychromatic Intensity (CME PIS) or 3-Wavelength Color (CME COL).
GWT green_wavelength_wt....w		
RGB Controls: Green Wavelength Weight	Associates a weight with the PSF computation at the specified wavelength. The green PSF used in the RGB simulation will be the sum of the PSFs computed for each of the wavelengths specified by GWL commands with the appropriate weights, specified by the GWT command.	1.0

Command Syntax		
Screen Control	Explanation	Default
BWL blue_wavelength_nm...w		
RGB Controls: Blue Wavelength	Specifies one or more wavelengths that are to be used in computing the blue PSF for the RGB simulation method. You can enter up to 20 blue wavelength values (BWL commands). The blue PSF is the linear combination of the PSFs at each of the wavelengths specified with the BWL command, appropriately weighted according to the wavelength weight. From the command line, the command syntax element ...w means that you can enter several wavelengths at once (BWL 650 546 450), or you can include the Wi Wi.j wavelength qualifier prior to the value (BWL W1 450). Note: The number of wavelengths is limited to 20 for red, 20 for green, and 20 for blue, allowing a total of 60 wavelengths to define the three spectral bands.	440 nm for CME RGB. Not used for Polychromatic Intensity (CME PIS) or 3-Wavelength Color (CME COL).
BWT blue_wavelength_wt...w		
RGB Controls: Blue Wavelength Weight	Associates a weight with the PSF computation at the specified wavelength. The blue PSF used in the RGB simulation will be the sum of the PSFs computed for each of the wavelengths specified by BWL commands with the appropriate weights, specified by the BWT command.	1.0

Defining Output Controls

To define output control settings for the 2D Image Simulation (IMS) option, choose the **Analysis > Diffraction > 2D Image Simulation** menu. The **2D Image Simulation** dialog box displays. Click the **Output Controls** tab, and define the controls on this tab as described in the table below:



Command Syntax		
Screen Control	Explanation	Default
ODP [Yes No]		
Display Object File	Displays the input object in the Image Simulation file viewer.	Yes.
IDP [Yes No]		
Display IMS Results	Displays the image simulation in the Image Simulation file viewer.	Yes.
PDP [Yes No]		
Display PSF Map	Creates a diagnostic plot of the PSF array that is displayed in the Image Simulation file viewer.	No.

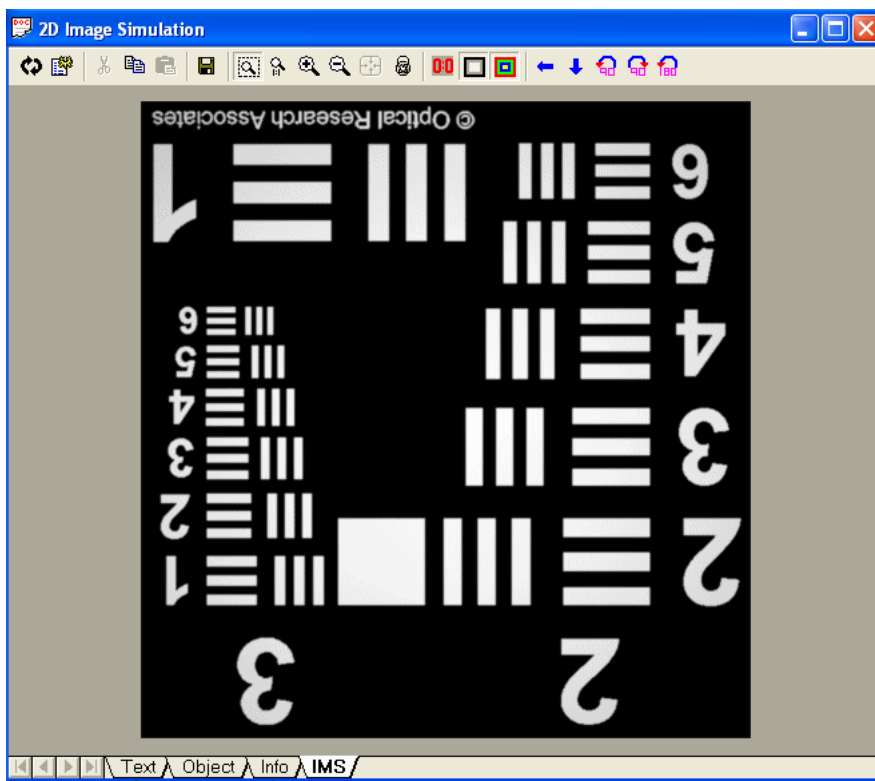
Command Syntax		
Screen Control	Explanation	Default
SVI [BMP JPG PNG] [COL GRY]		
Output file: File Type Color Scale	Automatically saves the image simulation output to a file in your CODE V working directory. The output filename is the input object filename, followed by an "_ims" extender (e.g., <i>usaf1951_ims.bmp</i>). Zoomed systems will generate a separate output file for each zoom position, where each filename will include a zoom position indicator (e.g., <i>usaf1951_ims_z1</i>, <i>usaf1951_ims_z2</i>, etc.). The default file format generated is a bitmap (BMP). You can also select JPEG (JPG) or Portable Network Graphic (PNG) file formats. If you are running IMS with the Polychromatic Intensity (CME PIS) computational method, the SVI command will only generate an 8-bit grayscale image file. If you are running IMS with the RGB (CME RGB) or 3-Wavelength Color (CME COL) computational methods, you can choose the following color types: • RGB (COL) generates a 24-bit color image file. • Grayscale (GRY) generates an 8-bit grayscale image file. For CME RGB, the file will contain the average R, G, and B value for each pixel to create the grayscale image. Note: If you do not use the SVI command, you can still save IMS output using the File > Save Window As menu when the IMS output window is in the foreground in the CODE V workspace. See “Using the Image Simulation File Viewer” on page 19-228 for details.	No file generated.

Command Syntax		
Screen Control	Explanation	Default
SID [Yes No]		
Output Data File	Saves image data to a 16- or 48-bit PNG file (16 bits/pixel/color). CODE V supplies the file name using the following conventions: the input object filename is followed by an “_imsdat” extender, followed by the .png file type (e.g., <i>usaf1951_imsdat.png</i>). Zoomed systems will generate a separate output file for each zoom position, where each filename will include a zoom position indicator (e.g., <i>usaf1951_imsdat_z1.png</i>, <i>usaf1951_imsdat_z2.png</i>, etc.). The SID command saves not only the image, but also any data that may be associated with the image (metadata), such as its title, simulation date, lens name, and object file name. Once saved, this format allows you to perform various image processing tasks on the results of an IMS simulation and display and/or save the results. Macro-PLUS predefined functions are available to perform image manipulation operations, such as adding and subtracting images, rotation, flipping, thresholding, and adding noise to an image. See “Image Simulation Functions” on page 25A-59. For a Macro-PLUS example using the SID command, see “Compute Image Contrast Using Image Simulation’s GetPixelValues Function” on page 25C-27, “Relative Illumination Correction Using Image Simulation Functions” on page 25C-30, and “Simulate a Three-Color Channel System Using Image Simulation Functions” on page 25C-45.	No.

Using the Image Simulation File Viewer

When you run 2D Image Simulation, the results will display in a file viewer that has several unique toolbar buttons to allow interactive image rotation, flip operations in X or Y, and toggling between color or grayscale (valid for 24-bit images only). The file viewer is a standard CODE V tabbed output window that includes tabs containing text output, informational messages, the input object image, and the simulation (output) image.

The following figure shows the 2D Image Simulation file viewer.



Click the tabs in the 2D Image Simulation file viewer to display its text output, any error or warning messages, the input object file, and the image simulation. In this figure, the image simulation is displayed (IMS tab).

When interpreting results from displayed graphic images, it is very important to understand issues related to the gamma associated with the file and display. See “Gamma Issues for Displaying, Printing, and Saving Files” on page 19-245. This information is especially important if the displayed object or IMS result appears overly bright or washed out.

In order to facilitate image orientation analysis, the initial IMS result is displayed relative to the image surface coordinate system with “Y” up. The “Z” axis will either be into or out of the screen. Light is always assumed to be traveling towards the observer.

File Viewer Toolbar Controls

The following table contains information about how the toolbar buttons in the 2D Image Simulation file viewer work. Some buttons that appear in the file viewer are not supported and are disabled. These disabled buttons are not documented in this section.

Toolbar Button	Description
 Execute	Runs IMS and regenerates the image simulation data in the same graphics window. Is useful when you have modified your optical system parameters and want to refresh the data so that it reflects your changes.
 Modify Settings	Displays the 2D Image Simulation dialog box, where you can change IMS settings and regenerate the simulation in the same graphics window.
 Copy	Copies the contents of the graphics window to the clipboard.
 Save Window	Saves the active window's contents with the current name, directory location, and file format. Invokes the Save As dialog box if you have not previously saved the window. Any orientation changes (i.e., rotations, flipping) made in the Image Simulation file viewer are saved in the output file, but zoomed-in views of the image are not saved. You can also save the active window's contents using the File > Save Window As menu. Note: To save all your lens data, use the Save Lens and Save Lens As options (File menu).
 Zoom	Activates the zoom mouse pointer so that you can magnify a selected area of the image simulation when you click the left mouse button and drag.
 Zoom 1:1	Fits the entire image inside the graphics window.
 Zoom In	Zooms in on the image.
 Zoom Out	Zooms out on the image.
 Center Image	Centers the image horizontally and vertically in the viewing window.
 Lock Zoom	Locks current zoom position for subsequent IMS graphics windows.
 Revert to Original Image	Reverts the image back to the original image orientation at 1x size.
 Convert to Grayscale	Converts image to 8-bit grayscale. Valid only for 24-bit color images.
 Revert to Color Image	Reconverts image to 24-bit color. Valid only when you have previously converted a 24-bit image to grayscale using the Convert to Grayscale button.
 Flip X	Flips the image along the X axis.

Toolbar Button	Description
 Flip Y	Flips the image along the Y axis.
 Rotate image 90 degrees counter-clockwise	Rotates the image orientation by 90 degrees in the counter-clockwise direction.
 Rotate image 90 degrees clockwise	Rotates the image orientation by 90 degrees in the counter-clockwise direction.
 Rotate by 180 degrees	Rotates the image orientation by 180 degrees.

Example Input Object Files Provided with 2D Image Simulation

Several example input object files are provided with CODE V in the **image** subdirectory of the CODE V main program directory:

C:\CODEV102\image

where C:\CODEV102 is the CODE V installation directory. For information about how to access and load example input object files in 2D Image Simulation, see “Defining the Input Object” on page 19-216.

These example image files are in bitmap (BMP) format. The examples supplied include:

black_dot_array.bmp	8-bit image containing a uniform grid of black circles on a white background.
black_f.bmp	8-bit image containing the letter F in black, on a white background.
color_circles_8bit.bmp	8-bit color image containing a uniform grid of colored circles on a black background.
color_circles_24bit.bmp	24-bit color image containing a uniform grid of colored circles on a black background.
default.bmp	Default bitmap image used by IMS. You can overwrite this file with your own bitmap image. The U.S. Airforce 1951 resolution target is installed by CODE V as the default.bmp file, but is also supplied as USAF1951_460kp.bmp.
EIA1956.bmp	8-bit image containing a television resolution target (i.e., EIA1956 chart).
grid1k_ur.bmp	8-bit image containing a white grid on a black background. The image dimensions are 1,024 x 1,024 pixels.
grid2k.bmp	8-bit image containing a white grid on a black background. The image dimensions are 2,048 x 2,048 pixels.
grid400.bmp	8-bit image containing a white grid on a black background. The image dimensions are 400 x 400 pixels.
grid.bmp	8-bit image containing a white grid on a black background. The image dimensions are 768 x 768 pixels.
Landscape_Mtn_Lake_1mp.bmp	24-bit color image with a landscape orientation (horizontal). One megapixel version.
Landscape_Mtn_Lake_7mp.bmp	24-bit color image with a landscape orientation (horizontal). Seven megapixel version.

Landscape_Mtn_Meadow_1mp.bmp	24-bit color image with a landscape orientation (horizontal). One megapixel version.
Landscape_Mtn_Meadow_7mp.bmp	24-bit color image with a landscape orientation (horizontal). Seven megapixel version.
Portrait_bear_2mp.bmp	24-bit color image with a portrait orientation (vertical). Two megapixel version.
Portrait_Puma_1mp.bmp	24-bit color image with a portrait orientation (vertical). One megapixel version.
Portrait_Puma_6mp.bmp	24-bit color image with a portrait orientation (vertical). Six megapixel version.
RGB_circles.bmp	24-bit color additive figure with overlapping red, green, and blue circles.
star_field.bmp	8-bit grayscale image of a star field.
USAF1951_8mp.bmp	1-bit image containing a U.S. Airforce 1951 resolution target (white bar targets on a black background). Eight megapixel version.
USAF1951_460kp.bmp	1-bit image containing a U.S. Airforce 1951 resolution target (white bar targets on a black background). 0.460 megapixel version.
USAF1951_array.bmp	8-bit image containing an array of U.S. Airforce 1951 resolution targets (white bar targets on a black background).
white_dot_array.bmp	8-bit image containing a uniform grid of white circles on a black background.
white_f.bmp	8-bit image containing the letter F in white, on a black background.
white_field_369kp.bmp	1-bit image that is useful for evaluating relative illumination across the field.

Discussion of Input and Computations

What to Include in the LDM Data

No additional LDM data is required.

Usage

The 2D Image Simulation (IMS) option simulates the appearance of a 2D input object as it is imaged through the optical system defined in CODE V. The input object is a bitmap (.BMP) file that can be sized to cover the full field, or a sub-area of the field. IMS includes the effects of distortion, aberration blurring, diffraction blurring, and relative illumination. Blurring due to detector pixel size can also be included. IMS output can be either a grayscale or RGB (red, green, and blue) image. To obtain the image simulation, CODE V computes an array of point spread function calculations (PSFs) from the optical system, which are convolved with the pixels of the input object. The PSFs can also be convolved with a rectangular detector to include the effect of detector blurring. Distortion is determined by chief ray tracing.

Default Assumptions of 2D Image Simulation

1. All zoom positions are included.

2. The field coordinate type is that of the system, determined by the type defined in the LDM (SPC command).
3. The default input object file is the U.S. Airforce resolution target (USAF1951.bmp).
4. The input object file is assumed to be centered at x and y field coordinates of (0,0) degrees.
5. The input object file is scaled to the maximum half-field in object space.
6. The size of the transform grid is 128 x 128.
7. The number of rays across the diameter is 64.
8. The PSF array size is 5 x 5 (number of samples in the x- and y-directions in the field at which the PSF is computed).
9. The PSF computation does not include the effects of detector blurring.
10. Will use the Polychromatic Intensity Analysis computational method, where IMS computes a single PSF that is a weighted combination of the monochromatic PSFs for each wavelength in the lens system. Wavelengths and their weights are those of the system, as defined in the LDM.
11. Output will be generated showing the input object file and image simulation. The PSF diagnostic image is not generated.

Input Object Definition

Controls on the **Object Definition** tab in the **2D Image Simulation** dialog box allow you to select an input object file as well as define the object's position, orientation, and scaling. See "Defining the Input Object" on page 19-216 for details about using these controls and the associated commands.

The **Object Filename** field (**OBJ** command) is where you specify an input bitmap file (.BMP format). This can be either an 8-bit or 24-bit color or grayscale file. CODE V supplies several example input files. See "Example Input Object Files Provided with 2D Image Simulation" on page 19-230 for details.

Controls are provided to define the object's position, orientation, and scaling with respect to one of the four field coordinate systems supported in CODE V. These include the ability to:

- Define the type of field coordinate (**Type of Field Coordinate**, or **FCO** command). By default, this is determined from the field specification type defined in the LDM.
- Size the object file in field units (**Field Semi-Diagonal**, or **OBH** command). By default, CODE V maps the half-diagonal of the object file to the maximum field specification of the lens system.
- Center the object file on an arbitrary X and Y field position (**X- and Y-Offsets**, or **XOF** and **YOF** commands). By default, CODE V assumes that the input object file is centered at X and Y field coordinates of (0,0) degrees.
- Orient the file using a gamma rotation that can be defined to cover a sub-area of the full object extent (**Rotation**, or **AZI** command). By default, CODE V will use the input object's original orientation. This gamma rotation gives you the flexibility to look at an input object at various angles, such as 45 degrees, and is particularly useful for resolution charts, which are normally oriented vertically and horizontally.

Point Spread Function Controls

To obtain the image simulation, CODE V computes an array of point spread functions (PSFs) from the optical system that are convolved with the pixels of the input image. Distortion effects are determined by chief ray tracing. The **PSF Controls** tab in the **2D Image Simulation** dialog box allow you to define the grid of PSFs to be computed over the object, and to enter the detector size to include blurring effects, if desired. See "Defining Point Spread Function (PSF) Controls" on page 19-219 for details about using the PSF controls and associated commands. Several standard PSF controls are provided, along with others specific to image simulation:

- The default size of the transform grid used in the FFT is 128. This may be modified (**FFT grid size** setting or **TGR**) but the value must be a power of 2 (minimum value is 16). If any other value is entered, it will be rounded up to the next allowable value. For example, **TGR 45** will result in a 64 x 64 transform grid. The default physical spacing in the image plane is selected so that the pupil grid at the shortest wavelength fills up half the transform grid. Use **TGR** to improve accuracy for systems with large or unusual obscurations, or to increase the size of the image area analyzed relative to the Airy disc.
- Due to the nature of the FFT process, if the pupil function is represented by many points, such as the default grid interval provides, the diffraction image (say, the Airy disc) will be represented by few points. Thus, asking for a smaller output grid spacing (**Focal plane grid increment** setting or **GRI**) to enlarge the image size will provide more detail in the output but will use less data to represent the lens (i.e., a smaller **Number of Rays Across Diameter** setting or **NRD**). This trade-off should be understood when choosing grid sizes. The default grid size should usually be the largest value used, while the smallest should be one-half of it. Values outside this range should be recognized as introducing increasingly large errors.
- You can specify how many different sample PSFs are computed and used in the image simulation with the **Number of Samples in X** and **Number of Samples in Y** controls (**PMX** and **PMY** commands). By default, the PSF array size is 5 x 5.
- You can override the system symmetry flag (**System Has Rotational Symmetry**, or **SYM** command), regardless of the actual symmetry. Forcing the computation to assume rotational symmetry has the advantage of facilitating a faster computation. By default, CODE V will use the symmetry state of the lens system.
- To include blurring due to a finite-sized detector, enter the rectangular detector dimensions using the **Detector Convolution** settings, or **DEX** and **DEY** commands. Note that this feature does not automatically downsample the image to show the detector structure, but will include the detector blurring in the PSF computation for the image simulation.

Color Controls

The **Color Controls** tab in the **2D Image Simulation** dialog box allows you to define the computational method that IMS uses to generate the output image. See “Defining Color Controls” on page 19-222 for details about using the color controls and associated commands. There are three computational methods that IMS uses to compute the output image:

- **Polychromatic Intensity** (**CME PIS** command) creates a polychromatic PSF for all system wavelengths and corresponding weights. The result is a single-band intensity image (8-bit grayscale image), which represents the intensity distribution across the image plane. If the input object is a 24-bit color image, each pixel takes on the average red, green, and blue values. This is the default method.
- **3-Wavelength Color** (**CME COL** command) maps the first (long), reference, and last (short) system wavelength with unity weights to red, green, and blue, respectively. The result is a 24-bit color image. You can also request grayscale output (24-bit, with the value of each pixel equal to the average red, green, and blue value for that pixel).
- **RGB** (**CME RGB** command) allows up to a 20-wavelength spectral band for red, green, and blue. Note that the wavelengths or wavelength ranges can cover any spectral band that is valid for the optical system. You can also apply wavelength weights. The default red, green, and blue wavelengths are 650 nm, 510 nm, and 440 nm, respectively, with unity weighting. The result is 24-bit color image. You can also request grayscale output (24-bit, with the value of each pixel equal to the average red, green, and blue value for that pixel).

Output Controls

The **Output Controls** tab in the **2D Image Simulation** dialog box allows you to define which types of output are displayed and whether an output file is automatically generated. See “Defining Output Controls” on page 19-225 for details about using these controls and associated commands.

The output controls include the ability to:

- Control whether the object file is displayed in the file viewer (**Display Object File**, or **ODP** command). By default, the object file is always displayed.
- Control whether the image simulation result is displayed in the file viewer (**Display IMS Results**, or **IDP** command). By default, the image simulation result is always displayed. Note that if you suppress the image simulation result, and have not specified that the result should be automatically saved in an output file, the IMS option will display an error message and terminate.
- Control whether the PSF diagnostic map is displayed in the file viewer (**Display PSF Map**, or **PDP** command). By default, the map is not displayed. When enabled, the PSF map will show:
 - The relative variation of PSFs across the defined image area, including red, green, and blue PSFs for CME COL or CME RGB.
 - The correct relative position of the PSFs to illustrate distortion.
 - The PSF intensities, normalized across the field of view, so that they are easily visible. Note that relative illumination information will not display.
- Control whether the image simulation output is automatically saved in a file (**Output Controls**, **SVI**, or **SID** commands). You are able to specify the type of file generated (bitmap, JPEG, or PNG), as well as the color type (RGB or grayscale). The file is saved in your CODE V working directory using a filename based on the input object filename, followed by an "_ims" extender. For example, if your input object filename is *24bit.bmp*, the output filename will be *24bit_ims.bmp*. Zoomed systems generate a separate output file for each zoom position, where each filename includes a zoom position indicator (e.g., *24bit_ims_z1*, *24bit_ims_z2*, etc.). Standard CODE V file versioning is used for duplicate filenames.

By selecting **Output Data File** (or **SID**), you save the image and its metadata to a 16- or 48-bit PNG file. The file is named in a similar manner to the SVI image file: the input object filename is followed by an "_imsdat" extender, followed by the .png file type (e.g., *usaf1951_imsdat.png*). Zoomed systems will generate a separate output file for each zoom position, where each filename will include a zoom position indicator (e.g., *usaf1951_imsdat_z1.png*, *usaf1951_imsdat_z2.png*, etc.). Once saved, this format allows you to perform various image processing tasks on the results of an IMS simulation and display and/or save the results. Macro-PLUS predefined functions are available to perform these image manipulation operations, such as adding and subtracting images, rotation, flipping, thresholding, and adding noise to an image. See "Image Simulation Functions" on page 25A-59.

The PNG file saved with the SID command will contain metadata to help identify the image and its origin. Those pieces of metadata that are not included in the basic PNG format (e.g., the image width and height) will all be written in a single text string. The metadata includes the following:

- Lens name
- Simulation date
- Object file name
- X width and Y height
- Normalization mode:
 - Normalized (NOR) implies that the peak has been scaled to equal 2^{16-1} in one channel for RGB images.
 - Raw (RAW) implies that the value of 256 (max value in the input BMP file) maps into a value of 2^{16-1} and no normalization is applied to the output IMS results.
- Scale value
 - For normalized (NOR) data, this is always 1.0. For RAW mode data, this is the scale value that should multiply the integer data in the file in order to produce the "true" image value. True value = (fileData - offsetValue)*scaleValue.
- Offset value:
 - For NOR mode, this is always 0. For RAW normalization mode, this has the meaning shown above

(the value 256 maps into a value of 2^{16-1}). It is necessary to allow unsigned integer data in the PNG file to represent negative as well as positive numbers. Data written directly from IMS will always have an offset value of 0. However, if Macro-PLUS is used to process the data and produce negative values, an offset and scale are needed before the data can be written to a file.

- Field units:
The field coordinates that are used in IMS (see the FCO command on page 19-217).
- Field size:
The object half-diagonal.
- Field offset X and Y
- Image semi-diagonal
- Field rotation:
Rotation applied to the input image (see the AZI command on page 19-218).
- Image pixel size:
X and Y pixel dimensions are identical, so only a single number is needed here.
- PSF inputs for simulation:
TGR (See the TGR command on page 19-219)
NRD or GRI (See the NRD command on page 19-220 and the GRI command on page 19-220)
PMX (See the PMX command on page 19-220)
PMY (See the PMY command on page 19-220)
DEX (See the DEX command on page 19-221)
DEY (See the DEY command on page 19-221)
Symmetry setting
- CME mode: PIS, RGB, or COL (See the CME command on page 19-222)
- Wavelengths and weights in a list:
For COL, there will be three wavelengths and three weights; for PIS, there can be any number of wavelengths and weights up to 21, and for RGB, there can be up to 21 wavelengths and weights for each channel. The data will have the following format in the metadata.

For COL:
WL RED value weight
WL GRN value weight
WL BLU value weight

For PIS:
WL PIS value weight
WL PIS value weight
... up to 21

For RGB:
WL RED value weight
WL RED value weight... up to 21 red wavelengths and weights
WL GRN value weight
WL GRN value weight... up to 21 green wavelengths and weights
WL BLU value weight
WL BLU value weight... up to 21 blue wavelengths and weights
- Gamma value
CODE V will always store results from IMS with a gamma value of 1.0. If the stored image is subsequently processed so that it has a different gamma, then the gamma due to this subsequent processing will be stored if the image is resaved. See “Gamma Issues for Displaying, Printing, and Saving Files” on page 19-245 for more detail on gamma issues.

Description of Output

The IMS output consists of the input object image, image simulation result, and PSF diagnostic map, as described in “Defining Output Controls” on page 19-225. IMS also produces text output for diagnostic purposes, which includes information about the object file attributes, computation controls, color controls, and image attributes. In addition, IMS provides status on its computation progress, since computation time can be significant, depending on the system and input object. Warnings and error are also displayed. For example, if you have entered a detector size, and the object file pixel is larger than the detector, CODE V will issue a warning. See “2D Image Simulation Errors and Warnings” on page 19-244 for examples of typical messages and how to address them.

Examples

Example 1. 2D Image Simulation of the Fisheye lens, showing distortion effects.

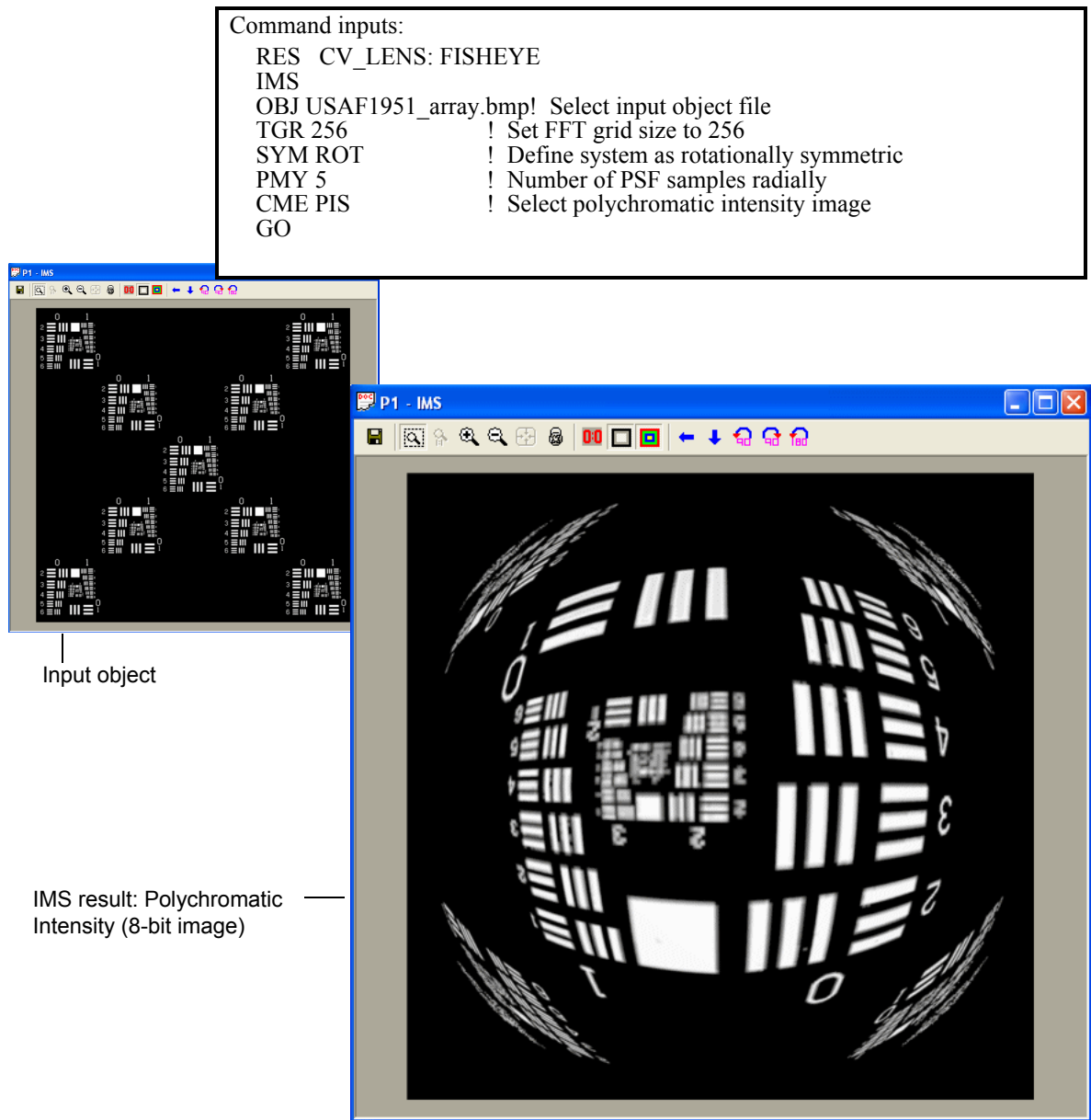


Figure 1. IMS input object file and output file showing distortion effects

Example 2. 2D Image Simulation of a thin lens module showing lateral color effects.

A thin lens module is used in the following example to show the effects of lateral color in IMS. The PSF diagnostic map is displayed. The lens parameters are as follows.

```
RDM N
LEN
TITLE 'Thin Lens Module      Aberration Generator'
FNO 1.0
DIM M
WL 650 510 440
REF 2
WTW 1 2 1
XIM 0.0 0.0
YIM 0.0 3.0
WTF 1.0 1.0
SO 0.0 1e9
S 0.0 0.0
STO
S 0.0 0.0
MOD
MFL 10.0
MED 5.0
MWL 650.0 510 440
MFD IMG 3.0
MCO C6 -30.0 ! LatColor W1
MCO C8 30.0 ! LatColor W3
MCO C14 0.0 ! Distortion
S 0.0 0.0
S 0.0 10.0
PIM
SI 0.0 0.0
GO
```

In this example, the input object, shown below, is defined to cover a sub-area of the full field of view (**Y-Offset** control, or **YOF** command). Note that the input object used in this example is not supplied with CODE V, but you can easily create your own pattern, such as this one, in a separate graphics program. Contact ORA Customer Support if you would like a copy of the input object file used in this example.

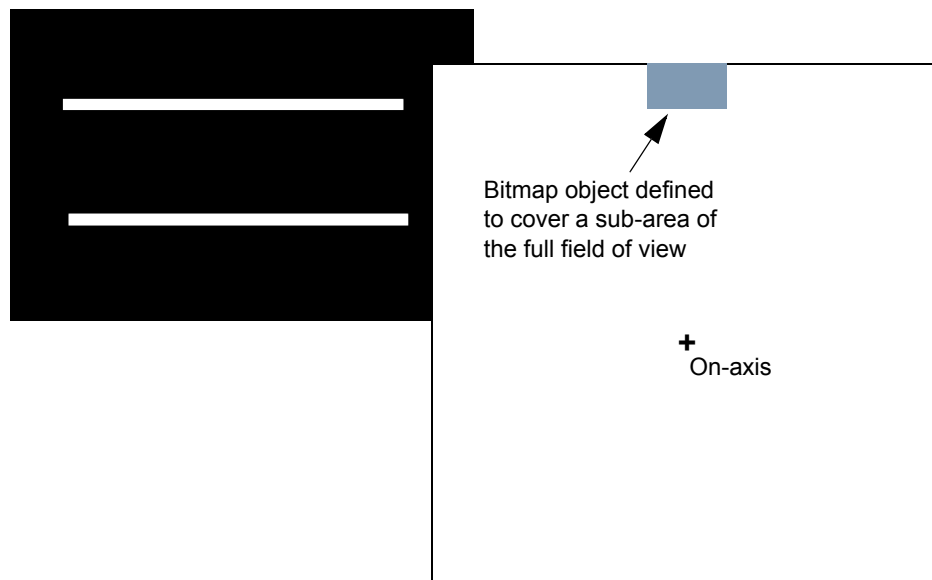


Figure 2. Input object file used in IMS lateral color example

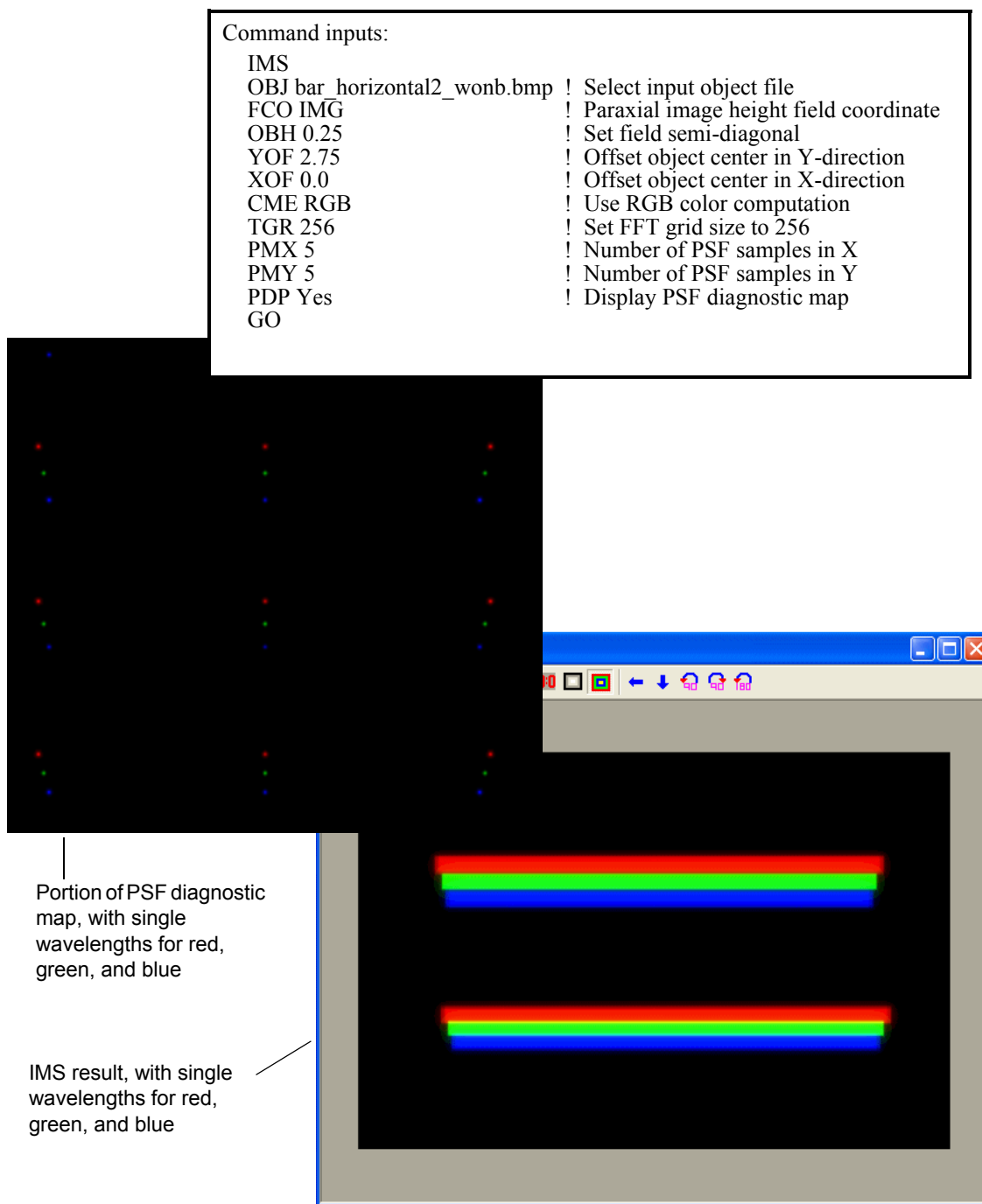


Figure 3. IMS output showing lateral color effects

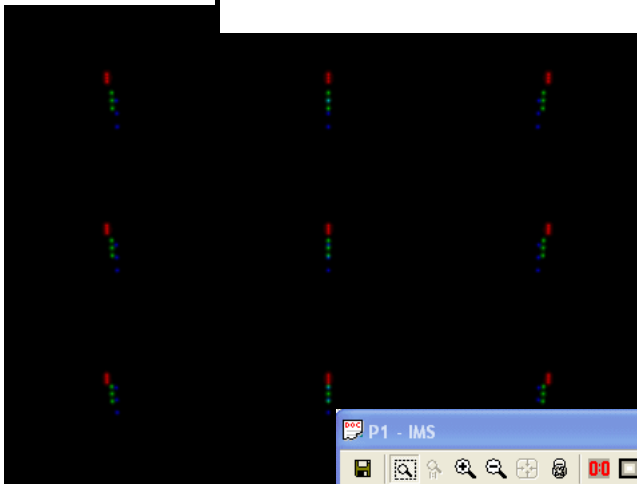
Figure 4 shows the same lens module system as above, but with the image simulation generated showing lateral color effects for multiple wavelengths.

Command inputs:

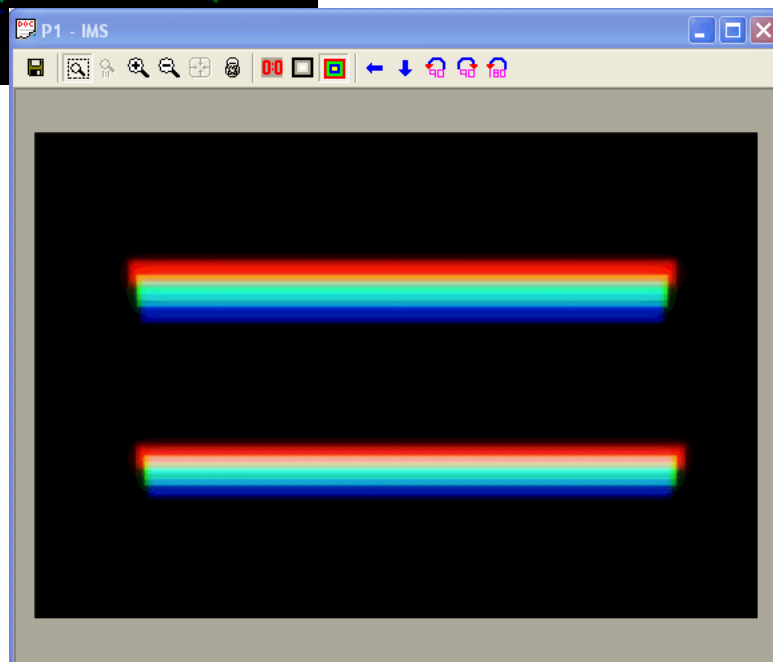
```

IMS
OBJ bar_horizontal2_wonb.bmp ! Select input object file
FCO IMG ! Paraxial image height field coordinate
OBH 0.25 ! Set field semi-diagonal
YOF 2.75 ! Offset object center in Y-direction
XOF 0.0 ! Offset object center in X-direction
CME RGB ! Use RGB color computation
RWL 650 645 640 ! Define red wavelengths
GWL 515 510 505 ! Define green wavelengths
BWL 450 445 440 ! Define blue wavelengths
TGR 256 ! Set FFT grid size to 256
PMX 5 ! Number of PSF samples in X
PMY 5 ! Number of PSF samples in Y
PDP Yes ! Display PSF diagnostic map
GO

```



Portion of PSF diagnostic map, with three wavelengths for each red, green, and blue value



IMS result, with three wavelengths for each red, green, and blue value

Figure 4. IMS output showing lateral color effects, with three wavelengths for each red, green, and blue value

Example 3. 2D Image Simulation of the Cooke triplet, with tilted object and image

The Cooke triplet with a tilted object and image is used in the following example to show a keystone effect in IMS. The lens parameters are as follows.

```
RDM;LEN
TITLE 'Cooke Triplet f/4.5'
FNO 4.5
DIM M
WL 656.3 546.1 486.1
REF 2
WTW 1 2 1
INI ' '
XAN 0.0 0.0 0.0
YAN 0.0 14.0000000977 20.0000001396
WTF 1.0 1.0 1.0
VUX 0.0 0.0 0.0
VLX 0.0 0.0 0.0
VUY 0.0 -0.014064157274 0.161710173349
VLY 0.0 0.068832339938 0.243553967992
SO 0.0 500.0
DAR
XDE 0.0; YDE 0.0; ZDE 0.0
ADE 45.0; BDE 0.0; CDE 0.0
S 21.48138 2.0 SK16_SCHOTT
CIR 7.0
S -124.1 5.26
S -19.1 1.25 F4_HOYA
STO
S 22.0 4.69
S 328.9 2.25 SK16_SCHOTT
CIR 6.5
S -16.7 48.4802441384
PIM
SI 0.0 0.0883504050234
THC 0
DAR
XDE 0.0; YDE 0.0; ZDE 0.0
ADE -6.26128192416; BDE 0.0; CDE 0.0; ADC 0; BDC 100; CDC 100
GO
```

The lens system is shown in Figure 5.

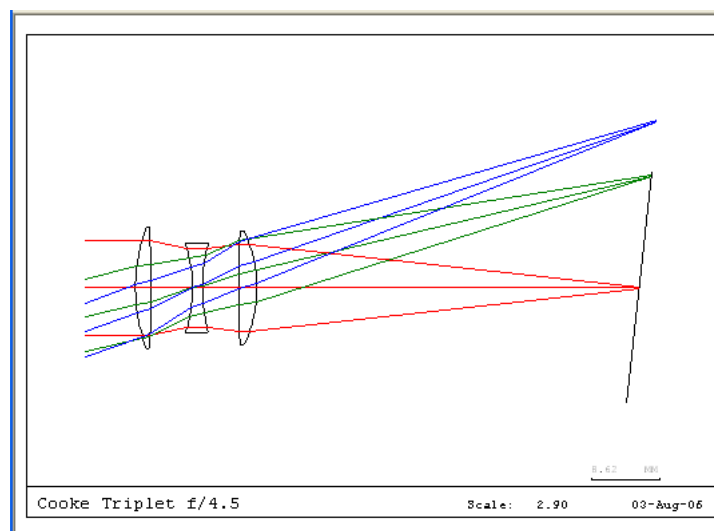


Figure 5. Cooke triplet with tilted object and image

Figure 6 shows a television resolution target imaged through the Cooke triplet.

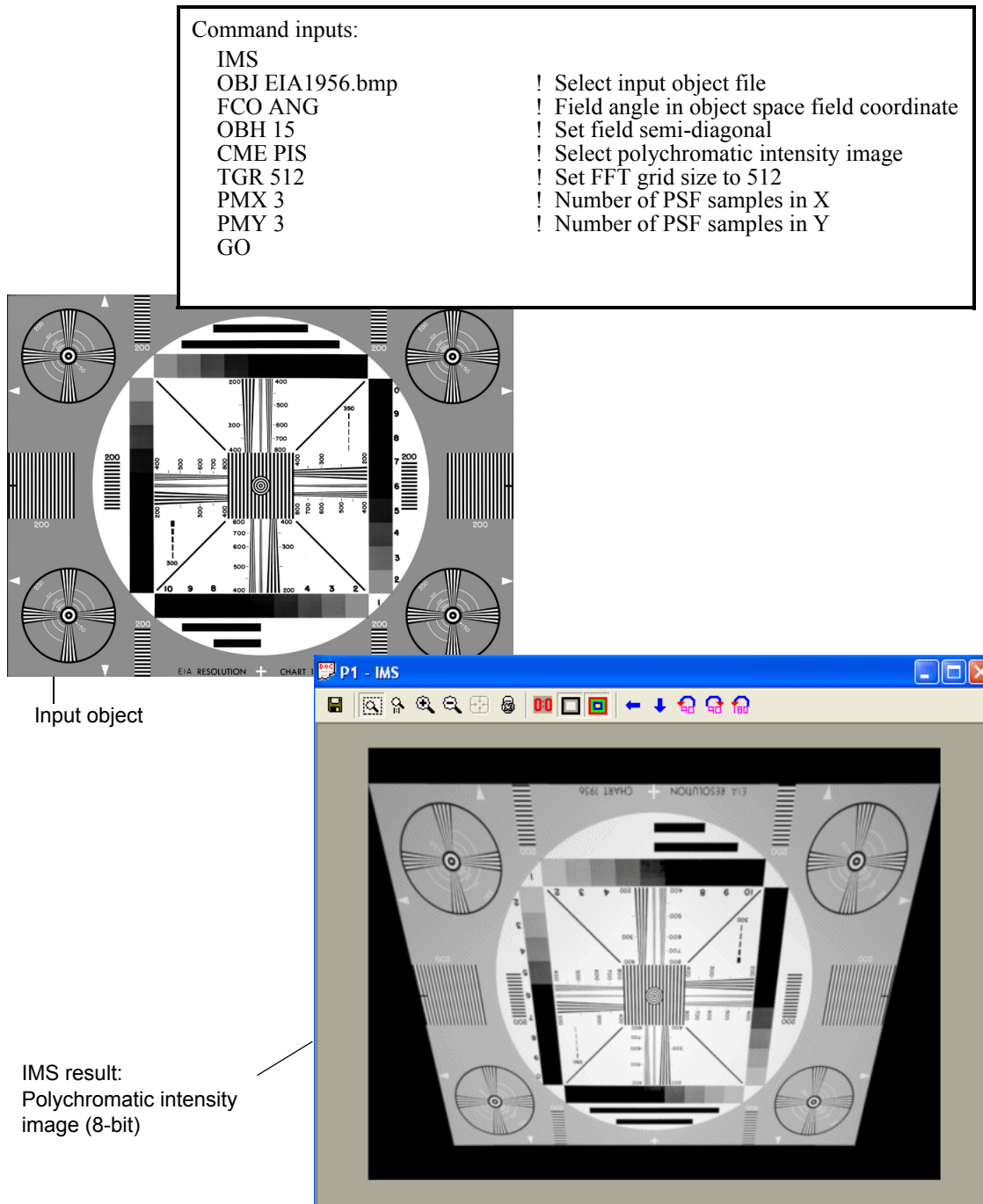


Figure 6. IMS input object file and output file showing keystone effect

Example 4. 2D Image Simulation of the Cooke triplet, showing best focus and defocus

The following example demonstrates how IMS can show aberrations in your lens system for a realistic object scene. To highlight IMS capabilities, this example first shows the original image, an image simulation through the F/4.5 Cooke triplet at best focus, and then an image simulation through a defocused Cooke triplet.

Original Object



Image through Cooke triplet at best focus



Command inputs for best focus:

```
RES CV_LENS: COOKE1
IMS
OBJ portrait_bear_2mp.bmp! Select input object file
TGR 256                ! Set FFT grid size to 256
PMX 5                  ! Number of radial PSF samples
CME RGB                ! Use RGB color computation
GO
```

Image through defocused Cooke triplet



Command inputs for defocus:

```
RES CV_LENS: COOKE1
TIN SI 2                ! Increment surface thickness
IMS
OBJ portrait_bear_2mp.bmp! Select input object file
TGR 512                ! Set FFT grid size to 512
PMX 5                  ! Number of radial PSF samples
CME RGB                ! Use RGB color computation
GO
```

Figure 7. IMS output showing Cooke triplet at best focus and defocus

**Example 5. 2D Image Simulation of the Cooke triplet showing detector blurring,
including images and listed output.**

Command inputs:

```
RES CV_LENS: COOKE1
IMS
OBJ portrait_bear_2mp.bmp! Select input object file
CME RGB ! Select RGB color computation method
TGR 512 ! Set FFT grid size to 512
PMX 3 ! Set number of PSF samples in X
PMY 3 ! Set number of PSF samples in Y
DEX .25 ! Set detector X-width
DEY .25 ! Set detector Y-width
GO
```



Input object

IMS result with 0.25 square
detector



Figure 8. IMS output showing detector blurring

IMS

1 zoom position is active

Object Attributes:

```
Input Object File:      c:\codev102\image\portrait_bear_2mp.bmp
Object File Type:       24-bit bitmap
Object File Resolution: 1377 x 1317
Field Coordinates:      Field Angle
Object Half-diagonal:   20.000000 degrees
Object Center (X,Y):    0.000000 , 0.000000 degrees
Object Rotation:        0.000000 degrees
```

IMS - 2D Image Simulation

Description of Output

```
Computation Controls:
Transform Grid Size (TGR):  512
Pupil Grid (NRD):          256
Image Grid (GRI)           0.001094 mm
PSF Array Size:            3 x 3
Rotational Symmetry:       No
Detector Size (X,Y):       0.250000 , 0.250000 mm

Color Controls:
Simulation Type:           RGB Analysis
R Wavelengths (nm) & Weights: 650.000000  1.000000
G Wavelengths (nm) & Weights: 510.000000  1.000000
B Wavelengths (nm) & Weights: 440.000000  1.000000

Output Controls:
Display Object File?:     Yes
Display IMS File(s)?:    Yes
Display PSF Diagnostic?:  No

Output IMS Folder:  C:\CVUSER
Output IMS File(s): None specified
Color Scale:       True Color

Image Attributes:
Image Semi-diagonal: 18.436237 mm
Image pixel size:    0.019351 x 0.019351 mm

PSF Computation Stage:
  25% complete...
  50% complete...
  75% complete...
 100% complete...
Image Simulation Stage:
 10% complete...
 25% complete...
 50% complete...
 75% complete...
100% complete...
```

2D Image Simulation Errors and Warnings

Typical IMS errors and warning messages, along with explanations, are listed in Appendix A, Error Messages and Troubleshooting. See “2D Image Simulation (IMS) Errors and Warnings” on page A-20 for details.

Technical Notes

Gamma Issues for Displaying, Printing, and Saving Files

Gamma is a parameter that controls how rendered luminance is mapped to display luminance. Understanding gamma is of particular importance for IMS in order to ensure that any relative illumination variations in the IMS result are rendered with qualitative accuracy, whether displayed or printed. The IMS algorithm assumes that the input object file has a linear relationship between pixel value and pixel brightness (i.e., a gamma of 1.0). The sample images that are provided with CODE V meet this requirement, but other images (e.g., from personal digital cameras, the Internet, scanners, etc.) often do not meet this requirement. These files will require gamma manipulation before they can be used accurately in IMS. Gamma manipulation, also called correction, can be achieved using the Gamma Correction Utility supplied with CODE V. See “Gamma Correction Utility” on page 19-248 for details. Or, you can use readily available commercial graphics manipulation software, such as Adobe Photoshop® or Corel Paint Shop Pro®. A description of how to apply gamma correction to graphics files from cameras or the Internet is described later in this section.

The gamma of a display is a measure of how the display renders brightness, as well as the ratio of red, green, and blue values. Gamma can be used to correct the non-linear relationship between pixel value and displayed intensity that is typical for a display device. This nonlinear relationship is roughly a power of function, where

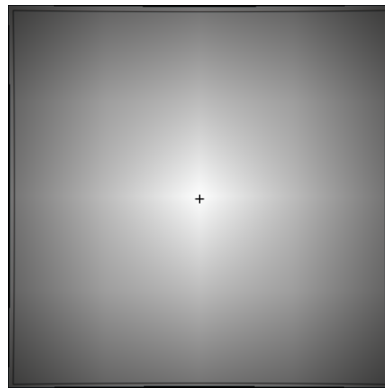
$$I_{\text{Display}} = (I_{\text{PixelValue}})^{\gamma}$$

Gamma correction is important for displaying an image accurately on different displays, projectors, and printers, since all these devices can display the same information differently.

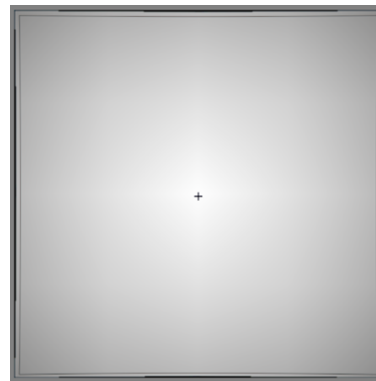
Most CRT monitors have an intensity-to-voltage response function of 2.5 power, requiring a gamma correction of 2.5 to render pixel brightness as expected (for an input file with an assumed linear relationship between pixel value and brightness). For example, if you display a .BMP file with pixel values of 127 and 254 on a CRT and measure the brightness difference, the pixel with a value of 254 will not appear twice as bright as the pixel with a value of 127. Instead, for a typical CRT, the 254-pixel will appear 5.7x as bright (i.e., a factor of $2^{2.5}$). Printers generally operate over a larger range of gamma values, but a gamma of 1.7 represents a reasonable average.

Bitmap (.BMP) and most other graphics files do not contain information regarding the gamma associated with the values in the file. For accurate results, the IMS algorithm requires that the pixel value be linear with brightness (an assumed gamma of 1.0). Therefore, all displayed IMS files (i.e., the original object and/or the IMS result) include a gamma correction of 2.5 so that the image brightness appears correct on the display. This gamma correction is retained when you use the Copy/Paste functions to capture the displayed image.

The following two figures show the effect of a 2.5 gamma correction in displayed IMS output. The figures show the resulting image brightness for an aberration-free system with a uniformly bright input field and 22% relative illumination in the corners of the image.



No gamma correction
(Gamma = 1.0)



Gamma corrected to 2.5

In the figures shown above, the appearance of the relative illumination fall-off from the center to the edge is qualitatively better for the gamma-corrected image when viewed on a computer display. The image that is not gamma corrected (i.e., Gamma = 1.0) appears to have a much larger brightness variation, which is not real. It is simply an artifact of the display response.

IMS results that are printed from CODE V are automatically gamma corrected to 1.7 for similar reasons.

As mentioned above, files created by other devices, such as digital cameras, scanners or downloaded from the Internet may have already been gamma corrected (i.e., scaled to have a non-linear relationship between pixel values and pixel brightness), so that they appear pleasing when displayed on a computer monitor. Files that have been previously gamma corrected to establish a non-linear relationship between pixel value and pixel brightness will typically appear overly bright or “washed-out” when displayed in IMS (which automatically applies an additional gamma correction of 2.5 on top of the prior gamma correction). In order to use these files accurately within IMS, the original gamma correction needs to be undone. This is accomplished by applying a new gamma correction, equal to $1/(\text{prior-}\gamma)$. If the prior gamma correction value of the file is unknown, a gamma correction of ~ 0.45 will work for most files generated by digital cameras or downloaded from the Internet. Use the Gamma Correction Utility provided in CODE V (see “Gamma Correction Utility” on page 19-248), or a commercial graphics manipulation package such as those mentioned above to apply the gamma correction. Files from digital cameras operating in “raw image” mode, or graphics files created in image manipulation software typically are not gamma corrected, and these files can generally be used with IMS without additional manipulation.

The following figures illustrate the appearance of a typical image from a digital camera and the gamma correction issues in using the image with the CODE V IMS option.



Picture from a Digital Camera: Relationship between pixel value and brightness is non-linear so that the image appears pleasing when displayed on a computer monitor.



Original Picture in CODE V: Accurate IMS results require that the input file have a linear relationship between pixel value and brightness. Since the original file did not, the gamma correction of 2.5 that IMS applies when displaying the result is in addition to the prior gamma correction and the result looks "washed-out" when displayed on a computer monitor.



Picture in Image Manipulation Software: After being gamma corrected by 0.45 to approximate the linear relationship between pixel value and pixel brightness (original gamma is unknown, so the correction may be approximate).



Original Digital Camera Picture with 0.45 Gamma Correction, as Displayed by CODE V: IMS results will be accurate for this input image, and displayed brightness variations will be representative of actual optical system performance.

All files saved from IMS (using the **SVI** command or the **File > Save Window As** menu), will not include any gamma correction. These files will retain the gamma assumption of the original file for the computed pixel values in the IMS result. This feature is useful if you want to manipulate these files in order to gamma-correct to a specific value based on a specific device (i.e., CRT monitor, LCD display, digital projector, specific printer, etc.).

Gamma Correction Utility

The Gamma Correction Utility provided with CODE V adjusts the pixel values of an input object file to a desired gamma value.

As described in “Gamma Issues for Displaying, Printing, and Saving Files” on page 19-245, the IMS algorithm assumes that the input object file has a linear relationship between pixel value and pixel brightness (i.e., a gamma of 1.0). All input object files supplied with CODE V (in the CV_IMAGE directory) have a gamma of 1.0. However, if you are using your own input object file, which may have previously been gamma corrected, then you will need to re-correct the file’s gamma value to 1.0 before using it in IMS. The Gamma Correction Utility is provided as a convenient way to perform this gamma correction. Output files saved by IMS also have a gamma of 1.0. You may wish to correct these files to a different gamma for display with a specific device. The Gamma Correction Utility can also be used for this purpose.

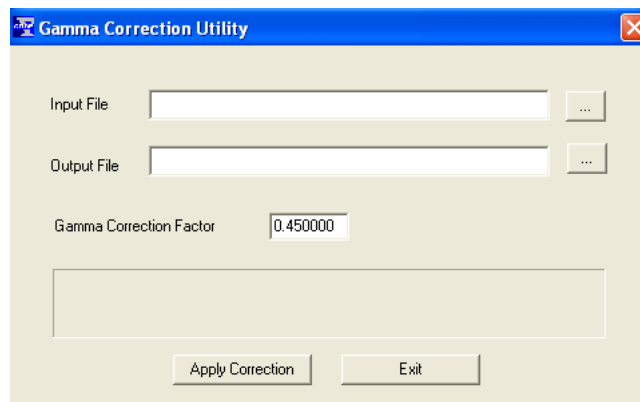
How to Use

To run the Gamma Correction Utility:

1. Click the Windows **Start** button, choose **Run**, and type:

C:\CODEV102\image\cvgamma.exe

where C:\CODEV102 is the CODE V installation directory. The **Gamma Correction Utility** dialog box is displayed.



2. Next to the **Input File** field, click the browse button [...] to navigate to and select your input object file.
3. In the **Output File** field, enter the filename for the gamma-corrected image. To save the gamma-corrected image in a different location than the input file, click the browse button [...].

If you do not enter a filename in the Output File field, CODE V will save the gamma correction in the original input file. CODE V will warn you before doing this, and provide you with an opportunity to cancel the operation and fill in an output filename.

4. Enter the **Gamma Correction Factor**.

The default value, 0.45, is based on the assumption that the input object has a gamma of 2.2, and needs to be corrected to a gamma of 1.0. A gamma of 2.2 is typical for digital camera images, or images downloaded from the Internet. If you do not know the prior gamma correction of your file, you can accept this default value. If you do know the gamma correction of your file, use the following formula to determine the correct gamma correction factor:

$$1/\text{prior gamma}$$

5. Click **Apply Correction** to complete the gamma correction and save the output file, or click **Exit** to cancel without saving the file.

Note that CODE V will apply an automatic gamma correction when displaying or printing IMS results. See “Gamma Issues for Displaying, Printing, and Saving Files” on page 19-245 for details.

Default RGB Wavelengths

There is no industry-standard correspondence of red, green, and blue colors (RGB) to wavelength. The CODE V default for RGB wavelength mapping was chosen based on a chart found at the following Web site:

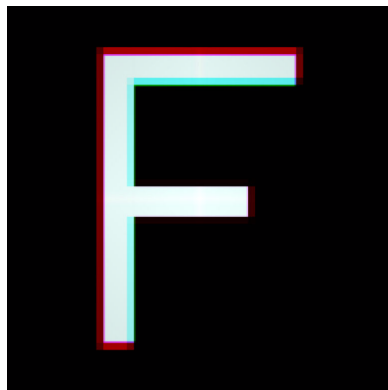
<http://webmineral.com/help/Reflectivity.shtml>

Typically, CODE V users will want to define multiple wavelengths for each color band in order to include polychromatic blurring in the PSF computation for each color.

Positive vs. Negative Object Files

It is important to recognize the difference between positive and negative object files when interpreting IMS results. This is particularly true for object files with 100% contrast differences. i.e., white objects on a field of black (positive objects) or black objects on a field of white (negative objects).

For example, in a system with lateral color, the color fringing will appear to be different for similar positive and negative objects (e.g., CV_IMAGE/white_F.bmp and CV_IMAGE/black_F.bmp).



Positive object



Negative object

Both results are correct. In the positive image, the object is the letter "F". In the negative image, the object is actually the square white field with a portion missing (corresponding to the black letter "F"). To correctly interpret the sign of the lateral color, it is easiest to evaluate the positive image, which shows a red letter "F" that is larger than the blue letter "F", which is negative lateral color.

Tradeoffs: Object File Resolution and Related Pixel Size at the Image

The computation time for IMS is directly related to the size (number of pixels) in the object file. Using a smaller object file will compute an IMS result more quickly. However, for the default file sizing where the semi-diagonal of the input file is mapped to the semi-maximum field of view of the lens, an input file with fewer pixels will mean that the effective pixel size at the image will be larger. In order to see meaningful results from IMS, it is important that the effective pixel size at the image is smaller than the detector pixel size (i.e., CCD or CMOS pixel, IR focal plane array detector size, the cone size of the human eye for visual systems, etc.). The IMS diagnostic output indicates the equivalent pixel size at the image for the object file. If you do not choose to define detector blurring in your IMS analysis, CODE V will also issue a warning that indicates the equivalent pixel size at the image, of the object file. You should check these numbers and verify that they are smaller than the intended detector for the system. If the pixel size at the image is larger than the intended detector, either use a higher resolution file to reduce the pixel size at the image, or define the current file over a sub-area of the field to achieve a smaller pixel size at the image. However, there is a lower limit, and the pixel size at the image must be at least twice as big as the transform grid sampling at the image.

Detector Convolution and Downsampling the Final Image to Match Detector Resolution

IMS allows you to optionally include the effect of blurring due to a finite detector size in the IMS result. Essentially, each computed PSF is convolved with a rectangular detector of a user-specified size before being convolved with each pixel in the object file. The convolution is performed for each monochromatic or polychromatic PSF generated by IMS, regardless of wavelength, and then convolved with every pixel in the object file. That is, there is no special processing based on detector structure (e.g., an RGB detector). If you enter a detector size, CODE V can compute the object file pixel size in image coordinates compared to the detector. A warning is issued if the pixel size is larger than the detector.

Currently, no downsampling is included in the IMS result. Downsampling is a means to show the detector structure. If the input object file is 1000x1000 pixels, but the detector is a CCD that is only 100x100 elements, the detected result for each CCD element will be based on the average brightness and color over 10x10 object file pixels. CODE V will optionally include the blurring based on the size of the CCD element, but the IMS result will still be based on 1000x1000 pixels.

Tradeoffs: Computation Time and Accuracy

IMS execution is performed in two stages. During the first stage, an array of PSFs is generated for the resident optical system. You can control the density of this PSF array and control the computation parameters for PSF (i.e., transform grid size, number of rays across the pupil, or image grid increment). In the default mode, a single, polychromatic PSF is generated for each PSF grid point. Alternatively, you can map one or more wavelengths to red, green, and blue channels such that three PSFs are computed for each PSF grid point. The two main computation time drivers during the PSF generation stage is the density of the PSF grid (i.e., PMX and PMY specification), and the execution time for a single PSF, which is primarily driven by the transform grid size (TGR).

The second stage of IMS execution is the convolution of each pixel in the object file with one or more PSFs from the PSF grid. For object file pixels that lie between PSF sample points, CODE V performs a linear interpolation between the nearest grid point PSFs to determine the convolution PSF. The primary computation time driver during this stage is the number of pixels in the object file, which directly relates to the number of convolutions that need to be performed.

The efficiency of the IMS calculation can be improved in the following ways:

- Specify rotational symmetry for PSF generation
- Use the minimum number of sample points in the PSF grid to achieve the desired accuracy
- Use the minimum transform grid size (TGR) for the PSF for accurate results
- Use an input object with the lowest resolution that still provides accurate results

If the system has rotational symmetry, a more efficient computation of the PSF grid is performed. A radial grid of PSFs corresponding to the maximum of either PMX or PMY is generated. CODE V recognizes rotational symmetry and will use this more efficient computation automatically. However, some systems may be rotationally symmetric, but CODE V does not recognize this (e.g., systems with fold mirrors). In this case, you can force PSF generation based on rotational symmetry using the ROT command.

The minimum number of sample points in the PSF grid to provide accurate results is dependent on both the system and how the object file is mapped. If only a small sub-portion of the field of view is being represented, a relatively sparse grid will be adequate (e.g., PMX = PMY = 3). For image orientation analyses where the blurring information is not important, use a single PSF (i.e., PMX = PMY = 1). The density of the PSF grid may need to be increased for wide-angle systems, or systems where there is significant variation in performance as a function of field of view.

The transform grid size (TGR) and image grid increment (GRI) should be chosen so that there are an adequate number of samples across the Airy disk and across the equivalent object file pixel size in the image, and so that no energy is detected at the edge of the transfer grid. CODE V will issue warnings if either of these conditions are violated. For systems with significant geometrical aberrations, the required TGR to suppress

all warnings about energy at the edge of the grid may be very large and require a long execution time. Remember that with the FFT process, energy outside the grid is aliased back into the grid. While this may impact the ultimate accuracy of the result, some aliasing may be acceptable and still provide a good qualitative understanding of the aberration blurring in the system.

A lower resolution input object mapped to the same field size as a higher resolution input object, will execute faster. However, you need to be aware of the object file resolution and related pixel size at the image tradeoffs discussed earlier. It may be more efficient to run IMS multiple times with a smaller input object file that sub-samples the full field of view.

