Getting Help in Vision Assistant

You can get help in Vision Assistant in any of the following ways:

- Refer to the Embedded Help window for context-sensitive help that applies to the currently selected tab or function.
- Select **Help»Online Help**.
- Move the mouse pointer over the buttons in the toolbar, Script window, Reference window, Acquisition window, or Image Browser for tooltips that describe what each button does.
- Select **Start»All Programs»National Instruments»Vision Builder AI»Documentation»NI Vision Concepts Manual** to launch the *NI Vision Concepts Manual*. This manual contains information about the algorithms, utilities, and tools available in Vision Assistant and the NI Vision library.

Processing an image

1. Load the image into the Processing window.
2. Select the appropriate function from the **Image**, **Color**, **Grayscale**, or **Binary** menu.

Tip You can run a particular step of a script by selecting the step in the Script window.

Related Topics

Magnifying an image

Printing an image

Selecting a region of interest (ROI)

1. Select an ROI tool from the toolbar.
2. Click and drag the tool on the image to select the ROI.

Note The ROI tools are accessible only for particular functions.

Tip If you want to select an ROI on an image and save it, click **Image»Image Mask** or select **Image Mask** in the **Image** tab of the Processing Functions palette. When you use the Broken Line and Polygon ROI tools, you must double-click to release the tool. When you release the mouse button after drawing an ROI, the ROI is green.

Related Topics

Tools Palette

Tools Palette

The tools palette is located at the top of the Processing window of Vision Assistant. You can use the tools palette to perform several functions, as described in the following list.

Print Image

Display image at 2:1 ratio and enlarge image further each time you click on the tool

Display image at 1:2 ratio and reduce image further each time you click on the tool

Return image to 100% view

Launch Vision Assistant online help

Display the color palette for the current image. You can change the palette to one of five predefined image palettes to show details not perceived with the current palette. Changing the palette here does not change the content of the image.

- **Gray Levels**—Gradual grayscale variation from black to white. Each value is assigned an equal amount of the RGB intensities.
- **Binary**—16 cycles of 16 different colors, where g is the grayscale value and g = 0 corresponds to R = 0, G = 0, B = 0 (black); g = 1 corresponds to R = 1, G = 0, B = 0 (red); g = 2 corresponds to R = 0, G = 1, B = 0 (green); and so on.
- **Gradient**—Gradation from red to white with a prominent range of light blue in the upper range. 0 is black and 255 is white.
- **Rainbow**—Gradation from blue to red with a prominent range of greens in the middle value range. 0 is black and 255 is white.
- **Temperature**—Gradation from light brown to dark brown. 0 is black and 255 is white.

Point tool—Click to select a point in an image.

Line tool—Click and drag to select pixels along that line.

Broken Line tool—Click and drag repeatedly to select a region. Double-click when you select the final point of the broken line to release the ROI tool.

Rectangle ROI tool—Click and drag to select a rectangular region. Hold down the <Shift> key while you click and drag to constrain the region to a square.

Rotated Rectangle ROI tool—Click and drag to select a rectangular region. Hold down the <Shift> key while you click and drag to contain the region to a square, and then click the lines inside the rectangle and drag to adjust the rotation angle.

Oval ROI tool—Click and drag to select an oval region. Hold down the <Shift> key while you click and drag to constrain the region to a circle.

Annulus ROI tool—Click the center position and drag to the size you want. Adjust the inner and outer radii and the start and end angles.

Polygon ROI tool—Click and drag repeatedly to select a polygon region. Double-click to complete the polygon.

Freehand ROI tool—Click and drag to select a region of the image. To select a region with this tool, verify that the ending point lies on the starting point to enclose a region.

Magic Wand tool—Create an image mask by extracting a region surrounding a reference pixel, called the origin, and using a tolerance (+ or –) of intensity variations based on this reference pixel. Using this origin, the Magic Wand Tool searches for its neighbors with an intensity equal to or falling within the tolerance value of the point of reference.

20 **Tolerance**—Set the maximum allowable deviation from the origin. All pixels satisfying the tolerance criteria (origin pixel – tolerance / origin pixel + tolerance) become part of the region.

Printing an image

1. Select the image that you want to print.
2. Click **File»Print Image**.
3. Type the title of the image and comments, if necessary. You can leave these options blank if you do not need them.
4. Check **Print Results Overlaid on the Image** to print overlay information such as templates you use for pattern matching and lines drawn with the Edge Detector function.
5. Click **Set Printer** to select a printer.
6. Click **OK**.

Related Topics

Magnifying an image

Zooming In and Out on Images

Use the **Zoom In** and **Zoom Out** tools to change the view of an image. The following magnification factors are displayed in the lower, left corner of the Processing window:

- 1/1 indicates 100% magnification (default)
- 2/1 indicates a 200% magnified view
- 1/2 indicates a 50% demagnified view

Reducing the magnification of an image

Click the **Zoom Out** tool located in the toolbar.

Tips

- Click the **Zoom In** or **Zoom 1:1** tool to return to the original magnification.
- Magnification factors are displayed in the lower, left corner of the processing window. 1/1 specifies 100% magnification (default). 2/1 specifies a 200% magnified view, and 1/2 specifies a 50% demagnified view.

Related Topics

Magnifying an image

Printing an image

Scripting

The Vision Assistant scripting feature records a series of image processing steps and the parameters you specified for each of those steps. You can run scripts on single images or image collections (batch processing). You also can modify and save scripts.

A script is a list of image processing and analysis functions and the parameters for each of those functions. As you prototype the image processing application, Vision Assistant records each function and its parameters.

Complete the following steps to develop a script:

1. Process an image.

   When you finish processing the image and exit the Parameter window, Vision Assistant adds the processing step to the current script.
2. Save the script.

You can use the script, which is also called a Builder file, to implement the image processing application using the NI Vision machine vision and image processing library in LabVIEW, LabWindows/CVI, Visual Basic, or other ActiveX containers.

If you have LabVIEW 6.1 or later and NI Vision 6.1 or later installed, you can use Vision Assistant to create the LabVIEW VI implementing the image processing steps you prototyped.

Related Topics

Creating a new script

Opening a script

Running a script

Saving a script

Script Window

Stepping through a script

Script Window

The Script window lists a series of image processing steps and the settings for each of those steps. The Script window contains several tools you can use to create, open, save, or run a script as well as remove or edit a step in the script. The Script window also contains tools that allow you to step through the script. As you step through the script or select any step in the script, the image you are currently processing changes to display the results of all image processing steps up to and including the selected step.

Related Topics

Opening a script

Running a script

Saving a script

Scripting

Stepping through a script

Creating a new script

You create a script when you process an image. You also can create a new script by clicking the **New** button in the Script window.

Related Topics

Adding a step to a script

Editing a step in a script

Opening a script

Removing a step from a script

Running a script

Saving a script

Stepping through a script

Opening a script

1. Click **File»Open Script** or click the **Open** button in the Script window.
2. Browse to the script you want to open.
3. Select the script.
4. Click **OK**.

Related Topics

Adding a step to a script

Editing a step in a script

Removing a step from a script

Running a script

Stepping through a script

Running a script

Click **Run Once** in the Script window. Running a script applies all steps in the script to the current image. If the last step is an analysis function, the results are automatically displayed.

Related Topics

Adding a step to a script

Editing a step in a script

Removing a step from a script

Stepping through a script

Saving a script

1. Click **File»Save Script** or click the **Save** button in the Script window.
2. Type the name of the script file.
3. Click **OK**.

Related Topics

Adding a step to a script

Editing a step in a script

Removing a step from a script

Running a script

Stepping through a script

Editing a step in a script

1. Double-click the step that you want to edit or select the appropriate step in the Script window and click **Edit Step**.
2. Change the function parameters in the Parameter window.
3. Click **OK**.

Tips

- If you modify the image and then decide that you do not want to save the changes, click **Cancel** in the Parameter window.
- You also can use **Edit Step** to modify the currently selected step.

Related Topics

Adding a step to a script

Removing a step from a script

Running a script

Saving a script

Stepping through a script

Adding a step to a script

You can insert a new step anywhere in an existing script.

1. In the script list, select the step that you want to precede the new step.
2. Select the image processing function that you want to apply.
3. Set the function parameters.
4. Click **OK**. Vision Assistant adds the step to the script below the selected step.

Related Topics

Editing a step in a script

Removing a step from a script

Running a script

Saving a script

Stepping through a script

Removing a step from a script

Select the step that you want to remove from the script and click **Edit»Cut** or right-click the step and select **Cut**.

Note Removing steps from a script can dramatically alter the final result of the script. As you delete steps, especially steps that change image type (such as conversion and color threshold), consider the how the rest of the script will run. The steps following the one you delete might try to process an image of an unexpected type, and you might receive unexpected results.

Tip You can use the **Previous Step** and **Next Step** buttons to step through the script. The image you are currently processing changes to reflect the effects of the function.

Related Topics

Adding a step to a script

Editing a step in a script

Running a script

Saving a script

Stepping through a script

Stepping through a script

Use the **Previous Step** and **Next Step** buttons in the Script window to step through the script. The image you are currently processing changes to reflect the effects of the function. You also can step through a script by clicking on the steps you want to view.

Related Topics

Adding a step to a script

Editing a step in a script

Removing a step from a script

Running a script

Performance Meter

You can use the Performance Meter to estimate how long (in milliseconds and frames/second) the script takes to complete in NI Vision on a given image.

Note The speed of some functions in the script varies depending on the image you process.

1. Open the script you want to test.
2. Select **Tools»Performance Meter**. Vision Assistant evaluates the performance of the functions in the script and displays an estimate of the total time NI Vision would take to complete the script.
3. Click **Details** to view an estimate of the time NI Vision would take to complete each function in the script.
4. Click **OK** to close the Performance Meter.

3D View (Image»3D View)

Displays the light intensity in a three-dimensional coordinate system, where the spatial coordinates of an image form two dimensions and the light intensity forms the third dimension. Each pixel from the image source is represented as a column of pixels in the 3D view. The pixel value corresponds to the altitude.

Note The 3D View function analyzes the image and displays information for you to review. This function does not modify the image.

3D View

- **Northwest**—Sets the viewing orientation for the 3D view in the Northwest direction.
- **Northeast**—Sets the viewing orientation for the 3D view in the Northeast direction.
- **Southwest**—Sets the viewing orientation for the 3D view in the Southwest direction.
- **Southeast**—Sets the viewing orientation for the 3D view in the Southeast direction.
- **Height**—Maximum height of a pixel drawn in 3D from the image source.
  - α —Angle between the horizontal axis and baseline. α must be between 0° and 45°.
  - β —Angle between the horizontal axis and second baseline. β must be between 0° and 45°.
- **Reduction**—Sets the size reduction. Reduction is a divisor that is applied to the image source when determining the final height and width of the 3D view image.
- **Invert**—Inverts the 3D view when selected.

Related Topics

Viewing an image in 3D

Viewing an image in 3D

Notes

- The 3D View function analyzes the image and displays information for you to review. This function does not modify the image.
- The 3D View function works only with 8-bit images.

1. Click **Image»3D View**, or select **3D View** in the **Image** tab of the Processing Functions palette.
2. Set the direction orientation.
3. Specify the maximum **Height** of a 3D pixel.
4. Specify the angle between the horizontal axis and baseline in α.
5. Define the angle between the horizontal axis and second baseline in β.
6. Set the size **Reduction**.
7. Click **OK** to close the Parameter window.

Tips Select **Invert** to invert the 3D image, which can be useful in distinguishing details in the darkest part of the image.

Related Topics

3D View (Image»3D View)

Brightness, Contrast, Gamma Correction (Image»Brightness)

Alters the brightness, contrast, and gamma of an image.

- **Brightness**—Brightness of the image. Brightness is expressed in gray levels, centered at 128. Higher values (up to 255) result in a brighter image, and lower values (down to 0) result in a darker image.
- **Contrast**—Contrast of the image, expressed as an angle. A value of 45° specifies no change in contrast. Higher values (up to 89°) increase contrast, and lower value (down to 1°) decrease contrast.
- **Gamma**—Gamma value used in Gamma Correction. The higher the Gamma coefficient, the stronger the intensity correction.
- **Lookup Table**—Displays the lookup table with the specified settings of Brightness, Contrast, and Gamma.
- **Color Plane**—Specifies the color plane to which you want to apply the transform.

  Note This parameter appears only when an RGB (color) image is loaded.
- **Reset**—Reverts **Brightness**, **Contrast**, and **Gamma** to neutral default settings.

Related Topics

Modifying brightness, contrast, and gamma values of an image

Modifying brightness, contrast, and gamma values of an image

Note The Brightness function works only with 8-bit images.

1. Click **Image»Brightness** or select **Brightness** in the **Image** tab of the Processing Functions palette.
2. Modify the **Brightness**, **Contrast**, and **Gamma** values with the slide or by manually entering the value. When you process a color image, you can apply **Brightness**, **Contrast**, and **Gamma** modifications to each **Color Plane** separately, or you can apply the same transformation to all color planes at the same time.
3. Click **OK** to add the step to the script.

Tips

- If you modify the values and want to return to the source image, click the **Reset** button.
- You can modify the brightness of a portion of an image by drawing an ROI.

Related Topics

Brightness, Contrast, Gamma Correction (Image»Brightness)

Color Operators (Color»Color Operators)

Performs arithmetic and logical operations on images.

- **Operators**

  Image Source Opens the original input image.

  Add Adds a constant or an image to the original input image. This is a pixel-by-pixel operation.

  Subtract Subtracts a constant or an image from the original input image. This is a pixel-by-pixel operation.

  Multiply Multiplies the original input image by a constant or another image. This is a pixel-by-pixel operation.

  Divide Divides the original input image by a constant or another image. This is a pixel-by-pixel operation.

  And Performs a logical AND operation between the original input image and a constant or another image stored in the buffer. This is a bit-wise operation.

  Not And Performs a logical NAND operation between the original input image and a constant or another image stored in the buffer. This is a bit-wise operation.

  Or Performs a logical OR operation between the original input image and a constant or another image stored in the buffer. This is a bit-wise operation.

  Not Or Performs a logical NOR operation between the original input image and a constant or another image stored in the buffer. This is a bit-wise operation.

  Exclusive Or Performs an Exclusive OR operation between the original input image and a constant or another image stored in the buffer. This is a bit-wise operation.

  Not Exclusive Or Performs an Exclusive NOR operation between the original input image and a constant or another image stored in the buffer. This is a bit-wise operation.

  Logical Difference Finds the logical difference between the original input image and a constant or another image stored in the buffer. This is a bit-wise operation.

Mask Replaces pixels in the original input image that correspond to pixels with value 0 in the image mask as 0. The image mask is an 8-bit image that specifies the regions in the original input image to be modified. This is a pixel-by-pixel operation.

- **Constant**—Constant value to be used in an operation, when selected.
- **Image**—Image from an active buffer to be used in an operation, when selected. You can select only those image buffers that contain 8-bit or RGB images of the same size as the image source. Click **Image»Image Buffer** or select **Image Buffer** in the **Image** tab of the Processing Functions palette to store an image in an empty image buffer.

Related Topics

Adding a constant to a color image

Adding an image to a color image

Comparing a color image with a constant

Comparing a color image with another image

Dividing a color image by a constant

Dividing a color image by another image

Multiplying a color image by a constant

Multiplying a color image by another image

Subtracting a constant from a color image

Subtracting an image from a color image

Using the Color Operators function

Using the Color Operators function

1. Click **Color»Color Operators** or select **Color Operators** in the **Color** tab of the Processing Functions palette.
2. Select one of the functions from the **Operators** list.
3. Select either the **Constant** or **Image** option. The **Constant** option uses a constant and the current image in the operation. The **Image** option uses the current image and an image stored in an image buffer.
4. Enter the constant or select the image buffer that contains the image you are adding to the source image or enter the constant value.
5. Click **OK** to add the step to the script and close the Parameters window.

Note You must already have an image stored in an image buffer before you can perform an operation using the current image and another image. The buffered image must be the same size as the source image and must be an 8-bit or RGB image.

Related Topics

Color Operators (Color»Color Operators)

Adding a constant to a color image

1. Click **Color»Color Operators** or select **Color Operators** in the **Color** tab of the Processing Functions palette.
2. Select **Add** from the **Operators** list.
3. Select the **Constant** option.
4. Set the constant value.
5. Click **OK** to add the constant to the image.

Related Topics

Adding an image to a color image

Color Operators (Color»Color Operators)

Comparing a color image with a constant

Comparing a color image with another image

Dividing a color image by a constant

Dividing a color image by another image

Multiplying a color image by a constant

Multiplying a color image by another image

Subtracting a constant from a color image

Subtracting an image from a color image

Using the Color Operators function

Adding an image to a color image

1. Click **Color»Color Operators** or select **Color Operators** in the **Color** tab of the Processing Functions palette.
2. Select **Add** from the **Operators** list.
3. Select the **Image** option.
4. Select the image buffer that contains the image you are adding to the source image**.**
5. Click **OK** to add the images together and add this step to the script.

Notes

- You must already have an image stored in an image buffer before you can add, subtract, multiply, or divide two images. Click **Image»Image Buffer** to store a copy of the processed image in a buffer. You can insert the **Image»Image Buffer** step anywhere in a script.
- The buffer image must be the same size as the source image and must an 8-bit or RGB image.

Related Topics

Adding a constant to a color image

Color Operators (Color»Color Operators)

Comparing a color image with a constant

Comparing a color image with another image

Dividing a color image by a constant

Dividing a color image by another image

Multiplying a color image by a constant

Multiplying a color image by another image

Subtracting a constant from a color image

Subtracting an image from a color image

Using the Color Operators function

Comparing a color image with a constant

1. Click **Color»Color Operators** or select **Color Operators** in the **Color** tab of the Processing Functions palette.
2. Select one of the logical operators from the **Operators** list.
3. Select the **Constant** option.
4. Set the constant value that you want to compare with the image.
5. Click **OK** to compare the constant with the image and add this step to the script.

Related Topics

Adding a constant to a color image

Adding an image to a color image

Color Operators (Color»Color Operators)

Comparing a color image with another image

Dividing a color image by a constant

Dividing a color image by another image

Multiplying a color image by a constant

Multiplying a color image by another image

Subtracting a constant from a color image

Subtracting an image from a color image

Using the Color Operators function

Comparing a color image with another image

1. Click **Color»Color Operators** or select **Color Operators** in the **Color** tab of the Processing Functions palette.
2. Select one of the logical operators from the **Operators** list.
3. Select the **Image** option.
4. Select the image buffer that contains the image you want to compare.
5. Click **OK** to compare the images and add this step to the script.

Notes

- You must already have an image stored in an image buffer before you can add, subtract, multiply, or divide two images. Click **Image»Image Buffer** to store a copy of the processed image in a buffer. You can insert the **Image»Image Buffer** step anywhere in a script.
- The buffer image must be the same size as the source image and must an 8-bit or RGB image.

Related Topics

Adding a constant to a color image

Adding an image to a color image

Color Operators (Color»Color Operators)

Comparing a color image with a constant

Dividing a color image by a constant

Dividing a color image by another image

Multiplying a color image by a constant

Multiplying a color image by another image

Subtracting a constant from a color image

Subtracting an image from a color image

Using the Color Operators function

Dividing a color image by a constant

1. Click **Color»Color Operators** or select **Color Operators** in the **Color** tab of the Processing Functions palette.
2. Select **Divide** from the **Operators** list.
3. Select the **Constant** option.
4. Set the constant value.
5. Click **OK** to divide the image by the constant.

Related Topics

Adding a constant to a color image

Adding an image to a color image

Color Operators (Color»Color Operators)

Comparing a color image with a constant

Comparing a color image with another image

Dividing a color image by another image

Multiplying a color image by a constant

Multiplying a color image by another image

Subtracting a constant from a color image

Subtracting an image from a color image

Using the Color Operators function

Dividing a color image by another image

1. Click **Color»Color Operators** or select **Color Operators** in the **Color** tab of the Processing Functions palette.
2. Select **Divide** from the **Operators** list.
3. Select the **Image** option.
4. Select the image buffer that contains the image by which you will divide the source image.
5. Click **OK** to divide the images and add this step to the script.

Notes

- You must already have an image stored in an image buffer before you can add, subtract, multiply, or divide two images. Click **Image»Image Buffer** to store a copy of the processed image in a buffer. You can insert the **Image»Image Buffer** step anywhere in a script.
- The buffer image must be the same size as the source image and must an 8-bit or RGB image.

Related Topics

Adding a constant to a color image

Adding an image to a color image

Color Operators (Color»Color Operators)

Comparing a color image with a constant

Comparing a color image with another image

Dividing a color image by a constant

Multiplying a color image by a constant

Multiplying a color image by another image

Subtracting a constant from a color image

Subtracting an image from a color image

Using the Color Operators function

Multiplying a color image by another image

1. Click **Color»Color Operators** or select **Color Operators** in the **Color** tab of the Processing Functions palette.
2. Select **Multiply** from the **Operators** list.
3. Select the **Image** option.
4. Select the image buffer that contains the image by which you will multiply the source image.
5. Click **OK** to multiply the images and add this step to the script.

Notes

- You must already have an image stored in an image buffer before you can add, subtract, multiply, or divide two images. Click **Image»Image Buffer** to store a copy of the processed image in a buffer. You can insert the **Image»Image Buffer** step anywhere in a script.
- The buffer image must be the same size as the source image and must an 8-bit or RGB image.

Related Topics

Adding a constant to a color image

Adding an image to a color image

Color Operators (Color»Color Operators)

Comparing a color image with a constant

Comparing a color image with another image

Dividing a color image by a constant

Dividing a color image by another image

Multiplying a color image by a constant

Subtracting a constant from a color image

Subtracting an image from a color image

Using the Color Operators function

Multiplying a color image by a constant

1. Click **Color»Color Operators** or select **Color Operators** in the **Color** tab of the Processing Functions palette.
2. Select **Multiply** from the **Operators** list.
3. Select the **Constant** option.
4. Set the constant value.
5. Click **OK** to multiply the image by the constant.

Related Topics

Adding a constant to a color image

Adding an image to a color image

Color Operators (Color»Color Operators)

Comparing a color image with a constant

Comparing a color image with another image

Dividing a color image by a constant

Dividing a color image by another image

Multiplying a color image by another image

Subtracting a constant from a color image

Subtracting an image from a color image

Using the Color Operators function

Subtracting a constant from a color image

1. Click **Color»Color Operators** or select **Color Operators** in the **Color** tab of the Processing Functions palette.
2. Select **Subtract** from the **Operators** list.
3. Select the **Constant** option.
4. Set the constant value.
5. Click **OK** to subtract the constant from the image.

Related Topics

Adding a constant to a color image

Adding an image to a color image

Color Operators (Color»Color Operators)

Comparing a color image with a constant

Comparing a color image with another image

Dividing a color image by a constant

Dividing a color image by another image

Multiplying a color image by a constant

Multiplying a color image by another image

Subtracting an image from a color image

Using the Color Operators function

Subtracting an image from a color image

1. Click **Color»Color Operators** or select **Color Operators** in the **Color** tab of the Processing Functions palette.
2. Select **Subtract** from the **Operators** list.
3. Select the **Image** option.
4. Select the image buffer that contains the image that you are subtracting from the source image.
5. Click **OK** to subtract the images and add this step to the script.

Notes

- You must already have an image stored in an image buffer before you can add, subtract, multiply, or divide two images. Click **Image»Image Buffer** to store a copy of the processed image in a buffer. You can insert the **Image»Image Buffer** step anywhere in a script.
- The buffer image must be the same size as the source image and must an 8-bit or RGB image.

Related Topics

Adding a constant to a color image

Adding an image to a color image

Color Operators (Color»Color Operators)

Comparing a color image with a constant

Comparing a color image with another image

Dividing a color image by a constant

Dividing a color image by another image

Multiplying a color image by a constant

Multiplying a color image by another image

Subtracting a constant from a color image

Using the Color Operators function

Conversion (Grayscale»Conversion)

Converts the current image to the specified image type.

- **From**—Image type for the current image.
- **To**—Image type that you want to convert the current image to. Valid values include 8-bit, 16-bit, and float.
- **Method**—Conversion method used to convert the image to a smaller data type.

  Adjust dynamic

  Adjusts the dynamic of the image so that the lowest intensity in the image corresponds to the smallest value available in the new image type and the highest value corresponds to the highest value available in the new image data type. All pixel intensities within those limits are distributed linearly.

  Note The Adjust dynamic function is not always available.

  Shift #

  Available only for 16-bit to 8-bit conversion. Performs Shift # to the 16-bit image prior to converting it to an 8-bit data type. The value of Shift # depends on the dynamic of the image. For example, if you acquired the image with a 12-bit camera and you want to convert it to an 8-bit image, keep the most significant bits. To keep the most significant bits in this case, perform 4 (12–8) shifts to retrieve the 8 most significant bits of the image.

  Cast

  Casts the values too large to be represented in the new data type to the largest value available and values too small to the smallest value available.

Related Topics

Converting grayscale images

Converting grayscale images

1. Click **Grayscale»Conversion** or click **Conversion** in the **Grayscale** tab of the Processing Functions palette.
2. Select the data type that you want to convert the current image to. If you are converting the image to a smaller data type, select the method of conversion.
3. Click **OK** to convert the image, add this step to the script, and close the Conversion window.

Related Topics

Conversion (Grayscale»Conversion)

Extract color planes (Color»Extract Color Planes)

Extracts one of the three color planes (Red, Green, Blue, Hue, Saturation, Luminance, Value) from an image.

Image Source	Original input image.
RGB: Red Plane	Extracts the red plane from an RGB image.
RGB: Green Plane	Extracts the green plane from an RGB image.
RGB: Blue Plane	Extracts the blue plane from an RGB image.
HSL: Hue Plane	Extracts the hue plane from an HSL image.
HSL: Saturation Plane	Extracts the saturation plane from an HSL image.
HSL: Luminance Plane	Extracts the luminance plane from an HSL image.
HSV: Value Plane	Extracts the value plane from an HSV image.
HSI: Intensity Plane	Extracts the intensity plane from an HSI image.

Related Topics

Extracting color planes from an image

Extracting color planes from an image

1. Click **Color»Extract Color Planes** or select **Extract Color Planes** in the **Color** tab of the Processing Functions palette.
2. Select the color plane that you want to extract.
3. Click **OK** to add this step to the script.

Note Because each color plane is made up of 8 bits, the color plane you extract is an 8-bit grayscale image.

Related Topics

Extract color planes (Color»Extract Color Planes)

FFT Filter (Grayscale»FFT Filter)

Applies a frequency filter to the image. This function performs three steps. First, it finds the fast Fourier transform (FFT) of the source image, which is a complex image. Then the function filters (truncates or attenuates) the complex image. Finally, it computes the inverse FFT. Generally, the FFT Filter is used to compute backgrounds in order to correct light drifts.

- **FFT Filtering**

Image Source	Opens the original input image.
Truncate	Truncates the frequencies of a complex image.
Attenuate	Attenuates the frequencies of a complex image.

- **Mode**—Determines which frequencies to truncate or attenuate.

Low Pass	Removes high frequencies.
High Pass	Removes low frequencies.

- **Truncation Frequency %**—Percentage of the frequencies retained within a Fourier-transformed image. This percentage is expressed with respect to the length of the diagonal of the FFT image and the **Mode** value.
- **Display FFT**—Shows the complex image after filtering.

Related Topics

Applying an FFT filter to an image

Applying an FFT filter to an image

1. Click **Grayscale»FFT Filter** or click **FFT Filter** in the **Grayscale** tab of the Processing Functions palette.
2. Select the Truncate or Attenuate option.
3. Select the **Mode**. **Low Pass** removes or attenuates high frequencies, and **High Pass** removes or attenuates low frequencies.
4. If you selected **Truncate**, set the **Truncation Frequency %**. Otherwise, proceed to the next step.
5. Click **OK** to apply the FFT Filter, add this step to the script, and close the Parameter window.

Note This function is computer-intensive and can take several seconds.

Tip If you want to see the complex image created by the FFT Filter, select the **Display FFT** option. To see details of the FFT Filter, select the Gradient color palette from the **Display Color Palette** control in the toolbar.

Related Topics

FFT Filter (Grayscale»FFT Filter)

Filters (Grayscale»Filters)

Prepares an image for processing so you can extract only the information you need from the image. Most of these filters apply a kernel across the image. A kernel represents a pixel and its relationship to neighboring pixels. The weight of the relationship is specified by the coefficients of each neighbor.

- **Filters**

  Image Source

  Original input image.

  Smoothing-Lowpass

  Lowpass filtering. Smoothes images by eliminating details and blurring edges.

  Smoothing-Local Average

  Local averaging of the image pixels based on the kernel.

  Smoothing-Gaussian

  Gaussian filtering based on the kernel. Attenuates the variations of light intensity in the neighborhood of a pixel. The Gaussian kernel has the following model: *a d c b x b c d a*

  where *a, b, c* and *d* are integers and *x* < 1.

  Smoothing-Median

  Median filtering. Each pixel is assigned the median value of its neighborhood.

  Edge Detection-Laplacian

  Laplacian filtering. Extracts the contour of objects and outlines details. The Laplacian filter kernel has the following model: *a d c b x b c d a*

  where *a, b, c,* and *d* are integers and *x* is greater than or equal to the sum of the absolute values of the outer coefficients.

  Edge Detection-Diff.

  Differentiation filtering. Produces continuous contours by highlighting each pixel where an intensity variation occurs between itself and its three upper left neighbors.

  Edge Detection-

  Prewitt filtering. A highpass filter that extracts the outer contours of objects.

Prewitt	
Edge Detection-Sobel	Sobel filter. A highpass filter that extracts the outer contours of objects.
Edge Detection-Roberts	Roberts filter to detect edges. Outlines the contours that highlight pixels where an intensity variation occurs along the diagonal axes.
Convolution-Highlight Details	Convolution kernel that highlights the edges of an image.
Convolution-Custom	Custom filtering using the kernel coefficients and size that you specify.

- **Kernel Size**—Size of the structuring element. Valid values include 3 x 3, 5 x 5, and 7 x 7.
- **Filter Size**—Sets the size of the filter for the Lowpass and Median functions.
- **Kernel**—Specifies the kernel coefficients.

Tip The kernel coefficients for each filter are set to default values. You may need to experiment with different coefficients and the kernel size to get the results you want.

Related Topics

Detecting edges with filters

Filtering an image with a custom convolution filter

Highlighting details in an image

Smoothing an image with filtering

Smoothing an image with filtering

1. Click **Grayscale»Filters** or click **Filters** in the **Grayscale** tab of the Processing Functions palette.
2. Select one of the smoothing functions from the **Filters** list.
3. Set the **Kernel Size** and coefficients if you are using a Local Average or Gaussian filter. Set the filter size if you are using the Lowpass or Median filter.
4. Click **OK** to filter the image and add this step to the script.

Tip Experiment with different coefficients and the **Kernel Size** to get the results you want.

Related Topics

Detecting edges with filters

Filling holes and smoothing right angles

Filtering an image with a custom convolution filter

Filters (Grayscale»Filters)

Highlighting details in an image

Smoothing object boundaries in binary images

Smoothing object boundaries in grayscale images

Detecting edges with filters

1. Click **Grayscale»Filters** or click **Filters** in the **Grayscale** tab of the Processing Functions palette.
2. Select one of the edge detection filters from the **Filters** list. If you are using the Laplacian filter, set the **Kernel Size** and coefficients.
3. Click **OK** to filter the image and add this step to the script.

Tip Experiment with different coefficients and the **Kernel Size** to get the results you want.

Related Topics

Filtering an image with a custom convolution filter

Filters (Grayscale»Filters)

Highlighting details in an image

Smoothing an image with filtering

Highlighting details in an image

1. Click **Grayscale»Filters** or click **Filters** in the **Grayscale** tab of the Processing Functions palette.
2. Select **Convolution-Highlight Details** from the **Filters** list.
3. Adjust the **Kernel Size** and coefficients, if necessary.
4. Click **OK** to filter the image and add this step to the script.

Tip Experiment with different coefficients and the **Kernel Size** to get the results you want.

Related Topics

Detecting edges with filters

Filtering an image with a custom convolution filter

Filters (Grayscale»Filters)

Smoothing an image with filtering

Filtering an image with a custom convolution filter

1. Click **Grayscale»Filters** or click **Filters** in the **Grayscale** tab of the Processing Functions palette.
2. Select **Convolution – Custom** from the **Filters** list.
3. Set the **Kernel Size** and coefficients.
4. Click **OK** to filter the image and add this step to the script.

Tip Experiment with different coefficients and the **Kernel Size** to get the results you want.

Related Topics

Detecting edges with filters

Filters (Grayscale»Filters)

Highlighting details in an image

Smoothing an image with filtering

Geometry (Image»Geometry)

Modifies the geometrical representation of an image.

- **Image Source**—Original input image.
- **Symmetry**—Transforms an image through its symmetry.
  - **Horizontal**—Horizontal axis of the image.
  - **Vertical**—Vertical axis of the image.
  - **Central**—Center of the image.
- **Rotation**—Rotates an image.
  - **Angle (degrees)**—Angle through which the image is rotated. To set the angle, type the angle in degrees or turn the knob.
- **Resampling**—Resamples an image based on user settings.
  - **X Resolution**—Image size in the x direction (in pixels).
  - **Y Resolution**—Image size in the y direction (in pixels).
  - **Keep Image Ratio**—Maintains the original ratio when selected.
  - **Interpolation**—Type of interpolation. Valid values include Zero Order, Bi-linear, Quadratic, or Cubic Spline.

Related Topics

Computing the symmetry of an image

Resampling an image

Rotating an image

Computing the symmetry of an image

1. Click **Image»Geometry** or select **Geometry** in the **Image** tab of the Processing Functions palette.
2. Select **Symmetry**.
3. Select a type of symmetry.
4. Click **OK** to add this step to the script.

Related Topics

Geometry (Image»Geometry)

Resampling an image

Rotating an image

Rotating an image

1. Click **Image»Geometry** or select **Geometry** in the **Image** tab of the Processing Functions palette.
2. Select **Rotation**.
3. Turn the knob to select the angle of rotation or manually enter an angle.
4. Click **OK** to add this step to the script.

Related Topics

Computing the symmetry of an image

Geometry (Image»Geometry)

Resampling an image

Resampling an image

1. Click **Image»Geometry** or select **Geometry** in the **Image** tab of the Processing Functions palette.
2. Select **Resampling**.
3. Select **Keep Image Ratio** to maintain the original image ratio. Deselect this option to change the image aspect ratio.
4. Set the **Interpolation Type**.
5. Set **X Resolution** and **Y Resolution**.
6. Click **OK** to add this step to the script.

Related Topics

Computing the symmetry of an image

Geometry (Image»Geometry)

Rotating an image

Get Image (Image»Get Image)

Opens an image from a file. Use this function when you need to open an image as a step in a script.

- **Path**—Browse to the location of the file you want to open.

Note You can use only the **Image»Get Image** function to configure a script to open an image.

Related Topics

Getting a new image

Getting an image

1. Click **Image»Get Image** or select **Get Image** in the **Image** tab of the Processing Functions palette.
2. Click the **Browse** button to navigate to the image you want the script to open.
3. Click **Open**.
4. Click **OK** to add this step to the script and close the Parameter window.

Note You can use only the **Image»Get Image** or select **Get Image** in the **Image** tab of the Processing Functions palette function to configure a script to open an image.

Related Topics

Get New Image (Image»Get New Image)

Get Image FAQs

Which image file formats are supported?

Vision Assistant supports the following image file formats: BMP, TIFF, JPEG, JPEG200, and PNG.

Is there a size limit for images I want to load?

Vision Assistant does not have an image size limit.

Operators (Grayscale»Operators)

Performs arithmetic and logical operations on images.

- **Operators**

  Image Source
  Opens the original input image.

  Add
  Adds a constant or an image to the original input image. This is a pixel-by-pixel operation.

  Subtract
  Subtracts a constant or an image from the original input image. This is a pixel-by-pixel operation.

  Multiply
  Multiplies the original input image by a constant or another image. This is a pixel-by-pixel operation.

  Divide
  Divides the original input image by a constant or another image. This is a pixel-by-pixel operation.

  And
  Performs a logical AND operation between the original input image and a constant or another image stored in the buffer. This is a bit-wise operation.

  Not And
  Performs a logical NAND operation between the original input image and a constant or another image stored in the buffer. This is a bit-wise operation.

  Or
  Performs a logical OR operation between the original input image and a constant or another image stored in the buffer. This is a bit-wise operation.

  Not Or
  Performs a logical NOR operation between the original input image and a constant or another image stored in the buffer. This is a bit-wise operation.

  Exclusive Or
  Performs an Exclusive OR operation between the original input image and a constant or another image stored in the buffer. This is a bit-wise operation.

  Not Exclusive Or
  Performs an Exclusive NOR operation between the original input image and a constant or another image stored in the buffer. This is a bit-wise operation.

  Logical Difference
  Finds the logical difference between the original input image and a constant or another image stored in the

	buffer. This is a bit-wise operation.
Mask	Replaces pixels in the original input image that correspond to pixels with value 0 in the image mask as 0. The image mask is an 8-bit image that specifies the regions in the original input image to be modified. This is a pixel-by-pixel operation.
Min	Extracts the smallest value between the original input image and a constant or another image stored in the buffer. This is a pixel-by-pixel operation.
Max	Extracts the largest value between the original input image and a constant or another image stored in the buffer. This is a pixel-by-pixel operation.

- **Constant**—Constant value to be used in an operation, when selected.
- **Image**—Image from an active buffer to be used in an operation, when selected. Vision Assistant allows you to select only image buffers that contain images of the same type and size as the image source. Click **Image»Image Buffer**, or select **Image Buffer** in the **Image** tab of the Processing Functions palette to store an image in an empty image buffer.

Related Topics

Adding a constant to a grayscale image

Adding a grayscale image to a grayscale image

Comparing a grayscale image with a constant

Comparing grayscale images

Dividing a grayscale image by a constant

Dividing a grayscale image by a grayscale image

Image Buffer (Image»Image Buffer)

Multiplying a grayscale image by a constant

Multiplying a grayscale image by a grayscale image

Retrieving an image from an image buffer

Storing an image in an image buffer

Subtracting a constant from a grayscale image

Subtracting a grayscale image from a grayscale image

Using the Operators function

Using the Operators function

1. Click **Grayscale»Operators** or click **Operators** in the **Grayscale** tab of the Processing Functions palette.
2. Select one of the functions from the **Operators** list.
3. Select either the **Constant** or **Image** option. The **Constant** option uses a constant and the current image in the operation. The **Image** option uses the current image and an image stored in an image buffer.
4. Enter the constant or select the image buffer that contains the image you are combining with the source image.
5. Click **OK** to perform the operation.

Note You must already have an image stored in an image buffer before you can perform an operation using the current image and another image. The buffered image must have the same size and type as the source image.

Related Topics

Operators (Grayscale»Operators)

Retrieving an image from an image buffer

Storing an image in an image buffer

Adding a constant to a grayscale image

1. Click **Grayscale»Operators** or click **Operators** in the **Grayscale** tab of the Processing Functions palette.
2. Select **Add** from the **Operators** list.
3. Select the **Constant** option.
4. Set the constant value.
5. Click **OK** to add the constant to the image.

Related Topics

Adding a grayscale image to a grayscale image

Comparing a grayscale image with a constant

Comparing grayscale images

Dividing a grayscale image by a constant

Dividing a grayscale image by a grayscale image

Image Buffer (Image»Image Buffer)

Multiplying a grayscale image by a constant

Multiplying a grayscale image by a grayscale image

Operators (Grayscale»Operators)

Retrieving an image from an image buffer

Storing an image in an image buffer

Subtracting a constant from a grayscale image

Subtracting a grayscale image from a grayscale image

Using the Operators function

Adding a grayscale image to a grayscale image

1. Click **Grayscale»Operators** or click **Operators** in the **Grayscale** tab of the Processing Functions palette.
2. Select **Add** from the **Operators** list.
3. Select the **Image** option.
4. Select the image buffer that contains the image you are adding to the source image**.**
5. Click **OK** to add the images together and add this step to the script.

Notes

- You must already have an image stored in an image buffer before you can add, subtract, multiply, or divide two images. Click **Image»Image Buffer** to store a copy of the processed image in a buffer. You can insert the **Image»Image Buffer** step anywhere in a script.
- The buffer image must be the same size and type as the source image.

Related Topics

Adding a constant to a grayscale image

Comparing a grayscale image with a constant

Comparing grayscale images

Dividing a grayscale image by a constant

Dividing a grayscale image by a grayscale image

Image Buffer (Image»Image Buffer)

Multiplying a grayscale image by a constant

Multiplying a grayscale image by a grayscale image

Operators (Grayscale»Operators)

Retrieving an image from an image buffer

Storing an image in an image buffer

Subtracting a constant from a grayscale image

Subtracting a grayscale image from a grayscale image

Using the Operators function

Averaging a grayscale image with a constant

1. Click **Grayscale»Operators** or click **Operators** in the **Grayscale** tab of the Processing Functions palette.
2. Select **Average** from the **Operators** list.
3. Select the **Constant** option.
4. Set the constant value that you want averaged with the image.
5. Click **OK** to average the constant with the image and add this step to the script.

Related Topics

Adding a constant to a grayscale image

Adding a grayscale image to a grayscale image

Averaging grayscale images

Comparing a grayscale image with a constant

Comparing grayscale images

Dividing a grayscale image by a constant

Dividing a grayscale image by a grayscale image

Image Buffer (Image»Image Buffer)

Multiplying a grayscale image by a constant

Multiplying a grayscale image by a grayscale image

Operators (Grayscale»Operators)

Retrieving an image from an image buffer

Storing an image in an image buffer

Subtracting a constant from a grayscale image

Subtracting a grayscale image from a grayscale image

Using the Operators function

Averaging grayscale images

1. Click **Grayscale»Operators** or click **Operators** in the **Grayscale** tab of the Processing Functions palette.
2. Select **Average** from the **Operators** list.
3. Select the **Image** option.
4. Select the image buffer that contains the image you want to use in the averaging.
5. Click **OK** to average the images and add this step to the script.

Notes

- You must already have an image stored in an image buffer before you can add, subtract, multiply, or divide two images. Click **Image»Image Buffer** to store a copy of the processed image in a buffer. You can insert the **Image»Image Buffer** step anywhere in a script.
- The buffer image must be the same size and type as the source image.

Related Topics

Adding a constant to a grayscale image

Adding a grayscale image to a grayscale image

Averaging a grayscale image with a constant

Comparing a grayscale image with a constant

Comparing grayscale images

Dividing a grayscale image by a constant

Dividing a grayscale image by a grayscale image

Image Buffer (Image»Image Buffer)

Multiplying a grayscale image by a constant

Multiplying a grayscale image by a grayscale image

Operators (Grayscale»Operators)

Retrieving an image from an image buffer

Storing an image in an image buffer

Subtracting a constant from a grayscale image

Subtracting a grayscale image from a grayscale image

Using the Operators function

Comparing grayscale images

1. Click **Grayscale»Operators** or click **Operators** in the **Grayscale** tab of the Processing Functions palette.
2. Select one of the logical operators from the **Operators** list.
3. Select the **Image** option.
4. Select the image buffer that contains the image you want to compare.
5. Click **OK** to compare the images and add this step to the script.

Notes

- You must already have an image stored in an image buffer before you can add, subtract, multiply, or divide two images. Click **Image»Image Buffer** to store a copy of the processed image in a buffer. You can insert the **Image»Image Buffer** step anywhere in a script.
- The buffer image must be the same size and type as the source image.

Related Topics

Adding a constant to a grayscale image

Adding a grayscale image to a grayscale image

Comparing a grayscale image with a constant

Dividing a grayscale image by a constant

Dividing a grayscale image by a grayscale image

Image Buffer (Image»Image Buffer)

Multiplying a grayscale image by a constant

Multiplying a grayscale image by a grayscale image

Operators (Grayscale»Operators)

Retrieving an image from an image buffer

Storing an image in an image buffer

Subtracting a constant from a grayscale image

Subtracting a grayscale image from a grayscale image

Using the Operators function

Comparing a grayscale image with a constant

1. Click **Grayscale»Operators** or click **Operators** in the **Grayscale** tab of the Processing Functions palette.
2. Select one of the logical operators from the **Operators** list.
3. Select the **Constant** option.
4. Set the constant value that you want to compare with the image.
5. Click **OK** to compare the constant with the image and add this step to the script.

Related Topics

Adding a constant to a grayscale image

Adding a grayscale image to a grayscale image

Comparing grayscale images

Dividing a grayscale image by a constant

Dividing a grayscale image by a grayscale image

Image Buffer (Image»Image Buffer)

Multiplying a grayscale image by a constant

Multiplying a grayscale image by a grayscale image

Operators (Grayscale»Operators)

Retrieving an image from an image buffer

Storing an image in an image buffer

Subtracting a constant from a grayscale image

Subtracting a grayscale image from a grayscale image

Using the Operators function

Dividing a grayscale image by a constant

1. Click **Grayscale»Operators** or click **Operators** in the **Grayscale** tab of the Processing Functions palette.
2. Select **Divide** from the **Operators** list.
3. Select the **Constant** option.
4. Set the constant value.
5. Click **OK** to divide the image by the constant.

Related Topics

Adding a constant to a grayscale image

Adding a grayscale image to a grayscale image

Comparing a grayscale image with a constant

Comparing grayscale images

Dividing a grayscale image by a grayscale image

Image Buffer (Image»Image Buffer)

Multiplying a grayscale image by a constant

Multiplying a grayscale image by a grayscale image

Operators (Grayscale»Operators)

Retrieving an image from an image buffer

Storing an image in an image buffer

Subtracting a constant from a grayscale image

Subtracting a grayscale image from a grayscale image

Using the Operators function

Dividing a grayscale image by a grayscale image

1. Click **Grayscale»Operators** or click **Operators** in the **Grayscale** tab of the Processing Functions palette.
2. Select **Divide** from the **Operators** list.
3. Select the **Image** option.
4. Select the image buffer that contains the image by which you will divide the source image.
5. Click **OK** to divide the images and add this step to the script.

Notes

- You must already have an image stored in an image buffer before you can add, subtract, multiply, or divide two images. Click **Image»Image Buffer** to store a copy of the processed image in a buffer. You can insert the **Image»Image Buffer** step anywhere in a script.
- The buffer image must be the same size and type as the source image.

Related Topics

Adding a constant to a grayscale image

Adding a grayscale image to a grayscale image

Comparing a grayscale image with a constant

Comparing grayscale images

Dividing a grayscale image by a constant

Image Buffer (Image»Image Buffer)

Multiplying a grayscale image by a constant

Multiplying a grayscale image by a grayscale image

Operators (Grayscale»Operators)

Retrieving an image from an image buffer

Storing an image in an image buffer

Subtracting a constant from a grayscale image

Subtracting a grayscale image from a grayscale image

Using the Operators function

Multiplying a grayscale image by a constant

1. Click **Grayscale»Operators** or click **Operators** in the **Grayscale** tab of the Processing Functions palette.
2. Select **Multiply** from the **Operators** list.
3. Select the **Constant** option.
4. Set the constant value.
5. Click **OK** to multiply the image by the constant.

Related Topics

Adding a constant to a grayscale image

Adding a grayscale image to a grayscale image

Comparing a grayscale image with a constant

Comparing grayscale images

Dividing a grayscale image by a constant

Dividing a grayscale image by a grayscale image

Image Buffer (Image»Image Buffer)

Operators (Grayscale»Operators)

Multiplying a grayscale image by a grayscale image

Retrieving an image from an image buffer

Storing an image in an image buffer

Subtracting a constant from a grayscale image

Subtracting a grayscale image from a grayscale image

Using the Operators function

Multiplying a grayscale image by a grayscale image

1. Click **Grayscale»Operators** or click **Operators** in the **Grayscale** tab of the Processing Functions palette.
2. Select **Multiply** from the **Operators** list.
3. Select the **Image** option.
4. Select the image buffer that contains the image by which you will multiply the source image.
5. Click **OK** to multiply the images and add this step to the script.

Notes

- You must already have an image stored in an image buffer before you can add, subtract, multiply, or divide two images. Click **Image»Image Buffer** to store a copy of the processed image in a buffer. You can insert the **Image»Image Buffer** step anywhere in a script.
- The buffer image must be the same size and type as the source image.

Related Topics

Adding a constant to a grayscale image

Adding a grayscale image to a grayscale image

Comparing a grayscale image with a constant

Comparing grayscale images

Dividing a grayscale image by a constant

Dividing a grayscale image by a grayscale image

Image Buffer (Image»Image Buffer)

Operators (Grayscale»Operators)

Multiplying a grayscale image by a constant

Retrieving an image from an image buffer

Storing an image in an image buffer

Subtracting a constant from a grayscale image

Subtracting a grayscale image from a grayscale image

Using the Operators function

Subtracting a constant from a grayscale image

1. Click **Grayscale»Operators** or click **Operators** in the **Grayscale** tab of the Processing Functions palette.
2. Select **Subtract** from the **Operators** list.
3. Select the **Constant** option.
4. Set the constant value.
5. Click **OK** to subtract the constant from the image.

Related Topics

Adding a constant to a grayscale image

Adding a grayscale image to a grayscale image

Comparing a grayscale image with a constant

Comparing grayscale images

Dividing a grayscale image by a constant

Dividing a grayscale image by a grayscale image

Image Buffer (Image»Image Buffer)

Operators (Grayscale»Operators)

Multiplying a grayscale image by a constant

Multiplying a grayscale image by a grayscale image

Retrieving an image from an image buffer

Storing an image in an image buffer

Subtracting a grayscale image from a grayscale image

Using the Operators function

Subtracting a grayscale image from a grayscale image

1. Click **Grayscale»Operators** or click **Operators** in the **Grayscale** tab of the Processing Functions palette.
2. Select **Subtract** from the **Operators** list.
3. Select the **Image** option.
4. Select the image buffer that contains the image that you are subtracting from the source image.
5. Click **OK** to subtract the images and add this step to the script.

Notes

- You must already have an image stored in an image buffer before you can add, subtract, multiply, or divide two images. Click **Image»Image Buffer** to store a copy of the processed image in a buffer. You can insert the **Image»Image Buffer** step anywhere in a script.
- The buffer image must be the same size and type as the source image.

Related Topics

Adding a constant to a grayscale image

Adding a grayscale image to a grayscale image

Comparing a grayscale image with a constant

Comparing grayscale images

Dividing a grayscale image by a constant

Dividing a grayscale image by a grayscale image

Image Buffer (Image»Image Buffer)

Operators (Grayscale»Operators)

Multiplying a grayscale image by a constant

Multiplying a grayscale image by a grayscale image

Retrieving an image from an image buffer

Storing an image in an image buffer

Subtracting a constant from a grayscale image

Using the Operators function

Histogram (Image»Histogram)

Counts the total number of pixels in each grayscale value and graphs it. You can perform a histogram on an entire image or selected regions of the image. Use the ROI tools from the toolbar to select regions in an image.

Note The Histogram function analyzes the image and displays information for you to review. This function does not modify the image.

- **Histogram**—Graphical representation of the quantitative distribution of pixels per grayscale value.
- **Minimum Value**—Smallest pixel intensity found in the image or selected regions in the image.
- **Maximum Value**—Largest pixel intensity found in the image or selected regions in the image.
- **Mean Value**—Mean value of intensities found in the image or selected regions in the image.
- **Std Deviation**—Standard deviation of the pixel intensities.
- **Number of Pixels**—Number of pixels used to calculate the histogram.
- **Mapping mode**—Specifies if the vertical scale (y-axis) of the histogram is linear or logarithmic.
    - **Linear**—Displays the histogram in a linear scale.
    - **Logarithmic**—Displays the histogram in a logarithmic scale. Use the logarithmic scale if the linear values are too small to view.

Related Topics

Performing a histogram

Selecting a region of interest (ROI)

Performing a histogram

1. Click **Image»Histogram** or select **Histogram** in the **Image** tab of the Processing Functions palette. Vision Assistant displays the histogram for the entire image.
2. If you want to perform a histogram for a portion of the image, select an ROI tool and draw an ROI on the image. The histogram displays data for the currently selected region.
3. Click **OK** to exit the Parameter window.

Note The Histogram function analyzes the image and displays information for you to review. This function does not modify the image.

Tips

- To display the histogram for the entire image after you selected an ROI or to delete the selected ROI, click the image or press the <Delete> key.
- If you want to save the histogram data, click the **Send Data to Excel** button or the **Save Results** button.

Related Topics

Histogram (Image»Histogram)

Selecting a region of interest (ROI)

Image Buffer (Image»Image Buffer)

Stores and retrieves images from buffers.

- **Current Image**—Lists image currently displayed in the Processing window.
- **Store Copy**—Adds the current image to an empty Image Buffer.
- **Retrieve Copy**—Retrieves the image from the selected Image Buffer.
- **Image Buffer**—Specifies the buffer to which an image is sent or from which an image is retrieved.

Related Topics

Retrieving an image from an image buffer

Storing an image in an image buffer

Storing an image in an image buffer

1. Click **Image»Image Buffer** or select **Image Buffer** in the **Image** tab of the Processing Functions palette.
2. Select **Store**.
3. Select an empty image buffer from the **Image Buffer** list. If you do not have an empty buffer, select a buffer that contains an image that you no longer need.
4. Click **OK** to save the image to the selected image buffer and add this step to the script.

Related Topics

Image Buffer (Image»Image Buffer)

Retrieving an image from an image buffer

Retrieving an image from an image buffer

1. Click **Image»Image Buffer** or select **Image Buffer** in the **Image** tab of the Processing Functions palette.
2. Select **Retrieve**.
3. Select the image buffer that contains the image you want to restore.
4. Click **OK** to restore the image and add this step to the script.

Note You must already have an image stored in an image buffer before you can restore it.

Related Topics

Image Buffer (Image»Image Buffer)

Storing an image in an image buffer

Image Mask (Image»Image Mask)

Builds a mask from an entire image or a region of interest (ROI). An image mask is an image containing values of 1 and 0. Vision Assistant processes only those pixels in the image source that have a corresponding mask image value of 1.

- **Image Mask**

Create from ROI	Creates an image mask from an ROI selected on the image.
Create from Image File	Creates an image mask from an image file.

- **Mask Options (Create from ROI)**

Outside the ROI	Sets all pixels outside the selected region to 0.
Inside the ROI	Sets all pixels inside the selected region to 0.
Extract Masked Region	Casts the resulting image to the bounding rectangle of the ROI. You can use this option with ROIs such as circles, ellipses, and freehand regions. When this option is not selected, the entire image is returned.

- **Mask Options (Create from Image File)**

Black in Image File	Masks all pixels with a corresponding mask image value equal to 0 (black).
Not Black in Image File	Masks all pixels not equal to 0 (non-black).
Extract Masked Region	Casts the resulting image to the bounding rectangle of the ROI. You can use this option with ROIs such as circles, ellipses, and freehand regions. When this option is not selected, the entire image is returned.

Related Topics

Masking an image from an image file

Masking an image from an ROI

Selecting a region of interest (ROI)

Masking an image from an ROI

1. Click **Image»Image Mask** or select **Image Mask** in the **Image** tab of the Processing Functions palette.
2. Select **Create from ROI**.
3. Select an ROI tool from the toolbar and draw the region of interest.
4. Click **Set ROI**.
5. Select mask options.
6. Click **OK** to add this step to the script.

Tip You can select a new region of interest by selecting an ROI tool and selecting a new area in the image.

Related Topics

Image Mask (Image»Image Mask)

Masking an image from an image file

Masking an image from an image file

1. Click **Image»Image Mask** or select **Image Mask** in the **Image** tab of the Processing Functions palette.
2. Select **Create from Image File**.
3. Click the **Browse** button, navigate to the location of the image file, select the file, and click **OK**.
4. Select mask options.
5. Click **OK** to add this step to the script.

Related Topics

Image Mask (Image»Image Mask)

Masking an image from an ROI

Selecting a region of interest (ROI)

Invert Binary Image (Binary»Invert Binary Image)

Reverses the dynamic of an image containing two different grayscale populations.

Notes

- To convert the binary image to an image containing pixel values 0 and 255, use the Multiply function to multiply the image by a constant value of 255 or use the Lookup Table:Equalize function to increase the dynamic of a binary image.
- Increase the dynamic of the binary image when you are applying arithmetic operators between the image and other images or constants.

Related Topics

Inverting a binary image

Inverting a binary image

Click **Binary»Invert Binary Image** or click **Invert Binary Image** in the **Binary** tab of the Processing Functions palette and click **OK** to reverse the dynamic of an image containing two different grayscale populations.

Related Topics

Invert Binary Image (Binary»Invert Binary Image)

Line Profile (Image»Line Profile)

Displays the grayscale distribution along a line of pixels in an image. Using the **Line Tool**, **Broken Line Tool**, or **Freehand Line Tool** from the toolbar, click and drag to draw a line segment.

Note The Line Profile function analyzes the image and displays information for you to review. This function does not modify the image.

- **Line Profile**—Variations of intensity along a line.
- **Minimum Value**—Smallest pixel intensity found in the line profile.
- **Maximum Value**—Largest pixel intensity found in the line profile.
- **Mean Value**—Mean value of pixel intensities found in the line profile.
- **Std Deviation**—Standard deviation of pixel intensities from the line profile.
- **Number of Pixels**—Number of pixels found in the line profile.
- **Mapping mode**—Specifies if the vertical scale (y-axis) of the histogram is linear or logarithmic.
    - **Linear**—Displays the line profile in a linear scale.
    - **Logarithmic**—Displays the line profile in a logarithmic scale. Use the logarithmic scale if the linear values are too small to view.

Related Topics

Performing a line profile

Selecting a region of interest (ROI)

Performing a line profile

1. Click **Image»Line Profile** or select **Line Profile** in the **Image** tab of the Processing Functions palette.
2. Select one of the line tools from the toolbar and draw a line on the image. The line profile displays data for the currently selected pixels.
3. Click **OK** to exit the Parameter window.

Note The Line Profile function analyzes the image and displays information for you to review. This function does not modify the image.

Tips

- Use the arrow keys (←, ↑, →, ↓) to move the line one pixel in the corresponding direction.
- To save the data, click the **Send Data to Excel** button or the **Save Results** button.

Related Topics

Line Profile (Image»Line Profile)

Selecting a region of interest (ROI)

Lookup Table (Grayscale»Lookup Table)

Improves contrast and brightness by applying a lookup table to an image. A lookup table (LUT) contains values used to transform the grayscale values of an image. For each grayscale value in the image, the corresponding new value is obtained from the LUT and assigned to every pixel of that grayscale value.

- **Lookup Table**

Image Source	Opens the original input image.
Equalize	Increases the intensity dynamic by distributing a given grayscale interval [min, max] over the full grayscale [0,255]. This function redistributes pixel intensities in order to provide a linear cumulated histogram.
Reverse	Reverses the pixel values, producing a photometric negative of the original image.
Logarithmic	Applies logarithmic transformation on the image pixels, which increases brightness and contrast in dark regions.
Exponential	Applies an exponential transformation on the image pixels, which decreases brightness and increases contrast in bright regions.
Square	Reduces contrast in dark regions. Similar to Exponential but with a more gradual effect.
Square Root	Reduces contrast in bright regions. Similar to Logarithmic but with a more gradual effect.
Power X	Reduces contrast in dark regions.
Power 1/X	Reduces contrast in bright regions.

- **Power Value**—Value of the power X for the Power X and Power 1/X functions. Default value is 1.5.

Related Topics

Performing a lookup table transformation

Performing a lookup table transformation

1. Click **Grayscale»Lookup Table** or click **Lookup Table** in the **Grayscale** tab of the Processing Functions palette.
2. Select the lookup table that you want to apply to the image.
3. Click **OK** to apply the lookup table and add this step to the script.

Related Topics

Lookup Table (Grayscale»Lookup Table)

Measure (Image»Measure)

Calculates measurement statistics associated with the region of interest (ROI).

Note The Measure function analyzes the image and displays information for you to review. This function does not modify the image.

Measurement

Types of measurement you can calculate, including position, length, angle, and area.

- **Position Results**

  Position of the point that you selected on the image using an ROI tool from the toolbar.

  - **X Pos.**—X-coordinate
  - **Y Pos.**—Y-coordinate
  - **Lev.**—Grayscale value of the pixel
- **Length Results**

  Measurements associated with the ROI.

  - **Length**—Length of the ROI in user-calibrated units.
  - **Start X**—X-coordinate of the starting point of the ROI.
  - **Start Y**—Y-coordinate of the starting point of the ROI.
  - **End X**—X-coordinate of the ending point of the ROI.
  - **End Y**—Y-coordinate of the ending point of the ROI.
  - **Mean**—Mean of the grayscale values found in the ROI.
  - **StdDev**—Standard deviation in the ROI.
  - **Min**—Smallest grayscale value found in the ROI.
  - **Max**—Largest grayscale value found in the ROI.
- **Angle Results**
  - **Angle**—Angle between any two line segments on the image, measured in degrees, counterclockwise.
- **Area Results**

  Measurements associated with the ROI.

  - **Area**—Area of the ROI.
  - **Ratio**—Ratio of the ROI area to the entire image area.
  - **X1**—Top, left x-coordinate of the ROI bounding rectangle.
  - **Y1**—Top, left y-coordinate of the ROI bounding rectangle.
  - **X2**—Bottom, right x-coordinate of the ROI bounding rectangle.

- **Y2**—Bottom, right y-coordinate of the ROI bounding rectangle.
- **Mean**—Mean of all grayscale values found in the ROI.
- **StdDev**—Standard deviation of all grayscale values found in the ROI.
- **Min**—Smallest grayscale value found in the ROI.
- **Max**—Largest grayscale value found in the ROI.

Related Topics

Calculating ROI measurements

Calculating ROI measurements

1. Click **Image»Measure** or select **Measure** in the **Image** tab of the Processing Functions palette.
2. Select the type of measurement that you want to calculate.
3. Draw an ROI on the image.

   Vision Assistant displays the results for that measurement in the **Results** table.

   Notes

   - The Measure function analyzes the image and displays information for you to review. This function does not modify the image.
   - Angles are measured in degrees, counterclockwise.

   Tips

   - To save the data, click the **Send Data to Excel** button or the **Save Results** button.
   - To remove the selected row, click **Delete Selection**. To delete all results, click **Delete All Results**.

4. Click **OK** to close the Parameter window.

Related Topics

Measure (Image»Measure)

Selecting a region of interest (ROI)

Morphological Functions

Use the following morphological functions to modify the shape of objects in an image.

Gray Morphology (Grayscale»Gray Morphology)

Dilate	Increases the brightness of each pixel surrounded by neighbors with a higher intensity.
Erode	Reduces the brightness of pixels that are surrounded by neighbors with a lower intensity.
Close	Removes dark spots isolated in bright regions and smoothes boundaries.
Open	Removes bright spots isolated in dark regions and smoothes boundaries.
Proper Close	Removes dark pixels isolated in bright regions and smoothes the boundaries of dark regions.
Proper Open	Removes bright pixels isolated in dark regions and smoothes the boundaries of bright regions.
Auto Median	Generates simpler objects with fewer details.

Related Topics

Expanding (dilating) objects in grayscale images

Filling gaps and holes (close) in grayscale images

Gray Morphology (Grayscale»Gray Morphology)

Reducing (eroding) objects in grayscale images

Removing small objects (open) from grayscale images

Smoothing object boundaries in grayscale images

Basic Morphology (Binary»Basic Morphology)

Erode objects
Eliminates pixels isolated in the background and erodes the contour of objects based on the structuring element. Erosion makes objects smaller.

Dilate objects
Eliminates tiny holes isolated in objects and expands the contour of the objects based on the structuring element. Dilation makes objects larger.

Open objects
Removes small objects and smoothes boundaries. An opening is an erosion followed by a dilation, which does not significantly alter the area or shape of objects. Borders removed by erosion are restored by dilation, but small objects that were absorbed during erosion do not reappear after dilation.

Close objects
Fills tiny holes and smoothes boundaries. A closing is a dilation followed by erosion, which does not significantly alter the area or shape of objects. Objects are expanded by the dilation and then reduced by erosion, so borders are smoothed and holes are filled.

Proper Open
Removes small particles and smoothes the contour of objects based on the structuring element. It is a finite and dual combination of openings and closings.

Proper Close
Fills tiny holes and smoothes inner contours of objects based on the structuring element. It is a finite and dual combination of closings and openings.

Gradient In
Extracts the interior contour of a particle.

Gradient Out
Extracts the exterior contour of a particle.

Auto Median
Simplifies objects based on the structuring element so that they have fewer details. The Auto Median function uses dual combinations of openings and closings.

Thick
Alters the shape of objects by adding parts to the object that match the pattern specified in the structuring element. Use thickening for filling holes and smoothing right angles along edges of objects.

Thin
Alters the shape of objects by eliminating parts of the object that match the pattern specified in the structuring element. Use thinning for removing single pixels isolated in the background and right angles along edges of objects.

Related Topics

Basic Morphology (Binary»Basic Morphology)

Expanding (dilating) objects in binary images

Extracting the contours of a particle

Filling gaps and holes (close) in binary images

Filling holes and smoothing right angles

Reducing (eroding) objects in binary images

Removing single pixels and right angles

Removing small objects (open) from binary images

Smoothing object boundaries in binary images

Adv. Morphology (Binary»Adv. Morphology)

Remove small objects	Removes small objects.
Remove large objects	Removes large objects.
Remove border objects	Eliminates particles that touch the borders of an image.
Fill holes	Fills the holes found in a particle. Holes are filled with a pixel value of 1.
Convex	Returns the convex hull of objects. You can use the convex hull of an object to make measurements when the object is amorphous (irregular).
Skeleton	Applies a succession of thinnings until the width of each object equals one pixel.
Separate objects	Breaks narrow isthmuses and separates objects that touch.
Label objects	Produces a labeled image using grayscale values equal to the number of objects in the image plus the grayscale 0 used in the background area.
Distance	Assigns to each pixel a grayscale value equal to the shortest distance to the border of the object, which might be to a hole within the object.
Danielsson	Creates a distance map similar to the Distance function. Danielsson uses a more accurate algorithm than Distance.
Segment image	Partitions a labeled image into segments, each centered on an object, so that the segments do not overlap and leave empty zones.

Related Topics

Adv. Morphology (Binary»Adv. Morphology)

Computing the convex hull of objects

Creating a distance map

Filling holes

Labeling objects

Reducing objects to their skeletons

Removing large objects

Removing objects touching the border of an image

Removing small objects

Segmenting an image

Separating objects

Applying morphological operations

1. Select one of the following morphology groups:
   - Click **Grayscale»Gray Morphology** or select **Gray Morphology** in the **Grayscale** tab of the Processing Functions palette.
   - Click **Binary»Basic Morphology** or select **Basic Morphology** in the **Binary** tab of the Processing Functions palette.
   - Click **Binary»Adv. Morphology** or click **Adv. Morphology** in the **Binary** tab of the Processing Functions palette.
2. Select one of the functions from the morphology list.
3. Set the parameters for the function, if needed. You might need to set parameters for the structuring element, connectivity, or pixel frame.
4. Click **OK** to apply the morphology and add this step to the script.

Related Topics

Adv. Morphology (Binary»Adv. Morphology)

Basic Morphology (Binary»Basic Morphology)

Computing the convex hull of objects

Creating a distance map

Expanding (dilating) objects in binary images

Expanding (dilating) objects in grayscale images

Extracting the contours of a particle

Filling gaps and holes (close) in binary images

Filling gaps and holes (close) in grayscale images

Filling holes

Filling holes and smoothing right angles

Gray Morphology (Grayscale»Gray Morphology)

Labeling objects

Reducing (eroding) objects in binary images

Reducing (eroding) objects in grayscale images

Reducing objects to their skeletons

Removing large objects

Removing objects touching the border of an image

Removing single pixels and right angles

Removing small objects

Removing small objects (open) from binary images

Removing small objects (open) from grayscale images

Segmenting an image

Separating objects

Smoothing object boundaries in binary images

Smoothing object boundaries in grayscale images

Gray Morphology (Grayscale»Gray Morphology)

Affects the structure of objects in grayscale images. You can use these functions for tasks such as expanding or reducing objects, filling holes, closing particles, or smoothing boundaries, which are tasks you perform to delineate objects and prepare images for thresholding and quantitative analysis.

- **Gray Morphology**

  Image Source
  Original input image.

  Dilate
  Grayscale dilation operation. Dilation increases the brightness of each pixel surrounded by neighbors with a higher intensity.

  Erode
  Grayscale erosion operation. Erosion reduces the brightness of pixels that are surrounded by neighbors with a lower intensity.

  Close
  Grayscale dilation followed by a grayscale erosion. Closing removes dark spots isolated in bright regions and smoothes boundaries.

  Open
  Grayscale erosion followed by a grayscale dilation. Opening removes bright spots isolated in dark regions and smoothes boundaries.

  Proper Close
  Finite and dual combination of closing and opening operations. Proper Closing removes dark pixels isolated in bright regions and smoothes the boundaries of dark regions.

  Proper Open
  Finite and dual combination of opening and closing operations. Proper Opening removes bright pixels isolated in dark regions and smoothes the boundaries of bright regions.

  Auto Median
  Dual combinations of openings and closings. Auto Median generates simpler objects with fewer details.

- **Structuring Element**

  ▦ 2D array used as a binary mask to define the neighbors of a pixel. You can modify the structuring element by clicking its cells. If a cell is black, it has a value of 1. If a cell is white (empty), it has a value of 0. If cell is black, the corresponding pixel is considered to be a neighbor and its value is used during the morphology operation.

- **Size**—Size of the structuring element. Valid values include 3 x 3, 5 x 5, and 7 x 7.
- **Iterations (Dilate and Erode functions)**—Number of times the operation is applied to the image.

Related Topics

Expanding (dilating) objects in grayscale images

Filling gaps and holes (close) in grayscale images

Reducing (eroding) objects in grayscale images

Removing small objects (open) from grayscale images

Smoothing object boundaries in grayscale images

Expanding (dilating) objects in grayscale images

1. Click **Grayscale»Gray Morphology** or select **Gray Morphology** in the **Grayscale** tab of the Processing Functions palette.
2. Select **Dilate** from the **Gray Morphology** list.
3. Set the **Size** of the **Structuring Element**.
4. Click the cells of the structuring element that you want to have a value of 0.
5. Enter the number of times that you want the dilation applied to the image in **Iterations**.
6. Click **OK** to apply the dilation and add this step to the script.

Tips

- The structuring element is a 2D array used as a binary mask to define the neighborhood of a pixel. You can modify the structuring element by clicking its cells. If a cell is black, it has a value of 1. If a cell is white (empty), it has a value of 0.
- Experiment with different coefficients and the **Kernel Size** to get the results you want.

Related Topics

Expanding (dilating) objects in binary images

Filling gaps and holes (close) in grayscale images

Gray Morphology (Grayscale»Gray Morphology)

Reducing (eroding) objects in grayscale images

Removing small objects (open) from grayscale images

Smoothing object boundaries in grayscale images

Reducing (eroding) objects in grayscale images

1. Click **Grayscale»Gray Morphology** or select **Gray Morphology** in the **Grayscale** tab of the Processing Functions palette.
2. Select **Erode** from the **Gray Morphology** list.
3. Set the **Size** of the **Structuring Element**.
4. Click the cells of the structuring element that you want to have a value of 0.
5. Enter the number of times that you want the erosion applied to the image in **Iterations**.
6. Click **OK** to apply the erosion and add this step to the script.

Tips

- The structuring element is a 2D array used as a binary mask. You can modify the structuring element by clicking its cells. If a cell is black, it has a value of 1. If a cell is white (empty), it has a value of 0.
- Experiment with different coefficients and the **Kernel Size** to get the results you want.

Related Topics

Expanding (dilating) objects in grayscale images

Filling gaps and holes (close) in grayscale images

Gray Morphology (Grayscale»Gray Morphology)

Reducing (eroding) objects in binary images

Removing small objects (open) from grayscale images

Smoothing object boundaries in grayscale images

Filling gaps and holes (close) in grayscale images

1. Click **Grayscale»Gray Morphology** or select **Gray Morphology** in the **Grayscale** tab of the Processing Functions palette.
2. Select **Close** from the **Gray Morphology** list.
3. Set the **Size** of the **Structuring Element**.
4. Click the cells of the structuring element that you want to have a value of 0.
5. Click **OK** to apply the closing and add this step to the script.

Tips

- The structuring element is a 2D array used as a binary mask to define the neighborhood of a pixel. You can modify the structuring element by clicking its cells. If a cell is black, it has a value of 1. If a cell is white (empty), it has a value of 0.
- Experiment with different coefficients and the **Kernel Size** to get the results you want.

Related Topics

Expanding (dilating) objects in grayscale images

Filling gaps and holes (close) in binary images

Gray Morphology (Grayscale»Gray Morphology)

Reducing (eroding) objects in grayscale images

Removing small objects (open) from grayscale images

Smoothing object boundaries in grayscale images

Removing small objects (open) from grayscale images

1. Click **Grayscale»Gray Morphology** or select **Gray Morphology** in the **Grayscale** tab of the Processing Functions palette.
2. Select **Open** from the **Gray Morphology** list.
3. Set the **Size** of the **Structuring Element**.
4. Click the cells of the structuring element that you want to have a value of 0.
5. Click **OK** to apply the opening and add this step to the script.

Tips

- The structuring element is a 2D array used as a binary mask to define the neighborhood of a pixel. You can modify the structuring element by clicking its cells. If a cell is black, it has a value of 1. If a cell is white (empty), it has a value of 0.
- Experiment with different coefficients and the **Kernel Size** to get the results you want.

Related Topics

Expanding (dilating) objects in grayscale images

Filling gaps and holes (close) in grayscale images

Gray Morphology (Grayscale»Gray Morphology)

Reducing (eroding) objects in grayscale images

Removing small objects (open) from binary images

Smoothing object boundaries in grayscale images

Smoothing object boundaries in grayscale images

1. Click **Grayscale»Gray Morphology** or select **Gray Morphology** in the **Grayscale** tab of the Processing Functions palette.
2. Select one of the three following morphological functions from the **Gray Morphology** list:
    - **Proper Open**—Smoothes boundaries of objects and fills small holes.
    - **Proper Close**—Smoothes boundaries of objects and removes small particles.
    - **Auto Median**—Smoothes boundaries of objects.
3. Set the **Size** of the **Structuring Element**.
4. Click the cells of the structuring element that you want to have a value of 0.
5. Click **OK** to apply the morphology and add this step to the script.

Tips

- The structuring element is a 2D array used as a binary mask to define the neighborhood of a pixel. You can modify the structuring element by clicking its cells. If a cell is black, it has a value of 1. If a cell is white (empty), it has a value of 0.
- Experiment with different coefficients and the **Kernel Size** to get the results you want.

Related Topics

Expanding (dilating) objects in grayscale images

Filling gaps and holes (close) in grayscale images

Filling holes and smoothing right angles

Gray Morphology (Grayscale»Gray Morphology)

Reducing (eroding) objects in grayscale images

Removing small objects (open) from grayscale images

Smoothing an image with filtering

Smoothing object boundaries in binary images

Basic Morphology (Binary»Basic Morphology)

Affects the shape of particles in binary images. Each particle or region is affected on an individual basis. You can use these functions for tasks such as expanding or reducing objects, filling holes, closing particles, or smoothing boundaries, which are tasks you perform to delineate objects and prepare images for quantitative analysis.

- **Basic Morphology**

Image Source	Opens the original input image.
Erode objects	Eliminates pixels isolated in the background and erodes the contour of objects based on the structuring element. Erosion makes objects smaller.
Dilate objects	Eliminates tiny holes isolated in objects and expands the contour of the objects based on the structuring element. Dilation makes objects larger.
Open objects	Removes small objects and smoothes boundaries. An opening is an erosion followed by a dilation, which does not significantly alter the area or shape of objects. Borders removed by erosion are restored by dilation, but small objects that were removed during erosion do not reappear after dilation.
Close objects	Fills tiny holes and smoothes boundaries. A closing is a dilation followed by erosion, which does not significantly alter the area or shape of objects. Objects are expanded by the dilation and then reduced by erosion, so borders are smoothed and holes are filled.
Proper Open	Removes small particles and smoothes the contour of objects based on the structuring element. It is a finite and dual combination of openings and closings.
Proper Close	Fills tiny holes and smoothes inner contours of objects based on the structuring element. It is a finite and dual combination of closings and openings.
Gradient In	Extracts the interior contours of a particle.

Gradient Out Extracts the exterior contours of a particle.

Auto Median Simplifies objects based on the structuring element so that they have fewer details. The Auto Median function uses dual combinations of openings and closings.

Thick Alters the shape of objects by adding parts to the object that match the pattern specified in the structuring element. Thickening is useful for filling holes and smoothing right angles along edges of objects.

Thin Alters the shape of objects by eliminating parts of the object that match the pattern specified in the structuring element. Thinning is useful for removing single pixels isolated in the background and right angles along edges of objects.

- **Structuring Element**

  ▦ 2D array used as a binary mask to define the neighbors of a pixel. You can modify the structuring element by clicking its cells. If a cell is black, it has a value of 1. If a cell is white (empty), it has a value of 0. If cell is black, the corresponding pixel is considered to be a neighbor and its value is used during the morphology operation.

- **Size**—Size of the structuring element. Valid values include 3 x 3, 5 x 5, and 7 x 7.
- **Iterations**—Specifies the number of 3 x 3 erosions to apply to the image.
- **Square/Hexagon**—Specifies the shape of the pixel frame during the transformation.

  ▦ Square pixel frame. Each pixel is surrounded by eight neighbors. The vertical and horizontal neighbors have a distance *d* from the pixel. Diagonal pixels have a slightly greater distance because they are farther away from the central pixel.

  ▦ Hexagon pixel frame. Each pixel is surrounded by six neighbors, which are each located at an equal distance from the central pixel.

Related Topics

Expanding (dilating) objects in binary images

Extracting the contours of a particle

Filling gaps and holes (close) in binary images

Filling holes and smoothing right angles

Reducing (eroding) objects in binary images

Removing single pixels and right angles

Removing small objects (open) from binary images

Smoothing object boundaries in binary images

Reducing (eroding) objects in binary images

1. Click **Binary»Basic Morphology** or select **Basic Morphology** in the **Binary** tab of the Processing Functions palette.
2. Select **Erode objects** from the **Basic Morphology** list.
3. Set the **Size** of the **Structuring Element**.
4. Click the cells of the structuring element that you want to have a value of 0.
5. Enter the number of times that you want the erosion applied to the image in **Iterations**.
6. Click **OK** to apply the erosion and add this step to the script.

Note The structuring element is a 2D array used as a binary mask. You can modify the structuring element by clicking its cells. If a cell is black, it has a value of 1. If a cell is white (empty), it has a value of 0.

Tip Experiment with different coefficients and the **Kernel Size** to get the results you want.

Related Topics

Basic Morphology (Binary»Basic Morphology)

Expanding (dilating) objects in binary images

Extracting the contours of a particle

Filling gaps and holes (close) in binary images

Filling holes and smoothing right angles

Reducing (eroding) objects in grayscale images

Removing single pixels and right angles

Removing small objects (open) from binary images

Smoothing object boundaries in binary images

Expanding (dilating) objects in binary images

1. Click **Binary»Basic Morphology** or select **Basic Morphology** in the **Binary** tab of the Processing Functions palette.
2. Select **Dilate objects** from the **Basic Morphology** list.
3. Set the **Size** of the **Structuring Element**.
4. Click the cells of the structuring element that you want to have a value of 0.
5. Enter the number of times that you want the dilation applied to the image in **Iterations**.
6. Click **OK** to apply the dilation and add this step to the script.

Note The structuring element is a 2D array used as a binary mask. You can modify the structuring element by clicking its cells. If a cell is black, it has a value of 1. If a cell is white (empty), it has a value of 0.

Tip Experiment with different coefficients and the **Kernel Size** to get the results you want.

Related Topics

Basic Morphology (Binary»Basic Morphology)

Expanding (dilating) objects in grayscale images

Extracting the contours of a particle

Filling gaps and holes (close) in binary images

Filling holes and smoothing right angles

Reducing (eroding) objects in binary images

Removing single pixels and right angles

Removing small objects (open) from binary images

Smoothing object boundaries in binary images

Removing small objects (open) from binary images

1. Click **Binary»Basic Morphology** or select **Basic Morphology** in the **Binary** tab of the Processing Functions palette.
2. Select **Open objects** from the **Basic Morphology** list.
3. Set the **Size** of the **Structuring Element**.
4. Click the cells of the structuring element that you want to have a value of 0.
5. Click **OK** to apply the opening and add this step to the script.

Note The structuring element is a 2D array used as a binary mask. You can modify the structuring element by clicking its cells. If a cell is black, it has a value of 1. If a cell is white (empty), it has a value of 0.

Tip Experiment with different coefficients and the **Kernel Size** to get the results you want.

Related Topics

Basic Morphology (Binary»Basic Morphology)

Expanding (dilating) objects in binary images

Extracting the contours of a particle

Filling gaps and holes (close) in binary images

Filling holes and smoothing right angles

Reducing (eroding) objects in binary images

Removing single pixels and right angles

Removing small objects (open) from grayscale images

Smoothing object boundaries in binary images

Filling gaps and holes (close) in binary images

1. Click **Binary»Basic Morphology** or select **Basic Morphology** in the **Binary** tab of the Processing Functions palette.
2. Select **Close objects** from the **Basic Morphology** list.
3. Set the **Size** of the **Structuring Element**.
4. Click the cells of the structuring element that you want to have a value of 0.
5. Click **OK** to apply the closing and add this step to the script.

Note The structuring element is a 2D array used as a binary mask. You can modify the structuring element by clicking its cells. If a cell is black, it has a value of 1. If a cell is white (empty), it has a value of 0.

Tip Experiment with different coefficients and the **Kernel Size** to get the results you want.

Related Topics

Basic Morphology (Binary»Basic Morphology)

Expanding (dilating) objects in binary images

Extracting the contours of a particle

Filling gaps and holes (close) in grayscale images

Filling holes and smoothing right angles

Reducing (eroding) objects in binary images

Removing single pixels and right angles

Removing small objects (open) from binary images

Smoothing object boundaries in binary images

Smoothing object boundaries in binary images

1. Click **Binary»Basic Morphology** or select **Basic Morphology** in the **Binary** tab of the Processing Functions palette.
2. Select a function from the **Basic Morphology** list. You can use one of three morphological functions to smooth the boundaries of objects:
    - **Proper Open**—Smoothes boundaries of objects and fills small holes.
    - **Proper Close**—Smoothes boundaries of objects and removes small particles.
    - **Auto Median**—Smoothes boundaries of objects.
3. Set the **Size** of the **Structuring Element**.
4. Click the cells of the structuring element that you want to have a value of 0.
5. Click **OK** to apply the smoothing function and add this step to the script.

Note The structuring element is a 2D array used as a binary mask. You can modify the structuring element by clicking its cells. If a cell is black, it has a value of 1. If a cell is white (empty), it has a value of 0.

Tip Experiment with different coefficients and the **Kernel Size** to get the results you want.

Related Topics

Basic Morphology (Binary»Basic Morphology)

Expanding (dilating) objects in binary images

Extracting the contours of a particle

Filling gaps and holes (close) in binary images

Filling holes and smoothing right angles

Reducing (eroding) objects in binary images

Removing single pixels and right angles

Removing small objects (open) from binary images

Smoothing object boundaries in grayscale images

Extracting the contours of a particle

1. Click **Binary»Basic Morphology** or select **Basic Morphology** in the **Binary** tab of the Processing Functions palette.
2. Select **Gradient In** to extract the interior contours of a particle or **Gradient Out** to extract the exterior contours of a particle.
3. Set the **Size** of the **Structuring Element**.
4. Click the cells of the structuring element that you want to have a value of 0.
5. Click **OK** to apply the function and add this step to the script.

Note The structuring element is a 2D array used as a binary mask. You can modify the structuring element by clicking its cells. If a cell is black, it has a value of 1. If a cell is white (empty), it has a value of 0.

Tip Experiment with different coefficients and the **Kernel Size** to get the results you want.

Related Topics

Basic Morphology (Binary»Basic Morphology)

Expanding (dilating) objects in binary images

Filling gaps and holes (close) in binary images

Filling holes and smoothing right angles

Reducing (eroding) objects in binary images

Removing single pixels and right angles

Removing small objects (open) from binary images

Smoothing object boundaries in binary images

Filling holes and smoothing right angles

1. Click **Binary»Basic Morphology** or select **Basic Morphology** in the **Binary** tab of the Processing Functions palette.
2. Select **Thick** to alter the shape of objects by adding parts to the object that match the pattern specified in the structuring element.
3. Set the **Size** of the **Structuring Element**.
4. Click the cells of the structuring element that you want to have a value of 0.
5. Click **OK** to apply the thickening and add this step to the script.

Note The structuring element is a 2D array used as a binary mask. You can modify the structuring element by clicking its cells. If a cell is black, it has a value of 1. If a cell is white (empty), it has a value of 0.

Tip Experiment with different coefficients and the **Kernel Size** to get the results you want.

Related Topics

Basic Morphology (Binary»Basic Morphology)

Expanding (dilating) objects in binary images

Extracting the contours of a particle

Filling gaps and holes (close) in binary images

Filling gaps and holes (close) in grayscale images

Filling holes

Reducing (eroding) objects in binary images

Removing single pixels and right angles

Removing small objects (open) from binary images

Smoothing an image with filtering

Smoothing object boundaries in binary images

Smoothing object boundaries in grayscale images

Removing single pixels and right angles

1. Click **Binary»Basic Morphology** or select **Basic Morphology** in the **Binary** tab of the Processing Functions palette.
2. Select **Thin** to alter the shape of objects by eliminating parts to the object that match the pattern specified in the structuring element.
3. Set the **Size** of the **Structuring Element**.
4. Click the cells of the structuring element that you want to have a value of 0.
5. Click **OK** to apply the thinning and add this step to the script.

Note The structuring element is a 2D array used as a binary mask. You can modify the structuring element by clicking its cells. If a cell is black, it has a value of 1. If a cell is white (empty), it has a value of 0.

Tip Experiment with different coefficients and the **Kernel Size** to get the results you want.

Related Topics

Basic Morphology (Binary»Basic Morphology)

Expanding (dilating) objects in binary images

Extracting the contours of a particle

Filling gaps and holes (close) in binary images

Filling holes and smoothing right angles

Reducing (eroding) objects in binary images

Removing small objects (open) from binary images

Smoothing object boundaries in binary images

Adv. Morphology (Binary»Adv. Morphology)

Perform high-level operations on particles in binary images. Use these functions for tasks such as removing small particles from an image, labeling particles in an image, or filling holes in particles.

- **Advanced Morphology**

Image Source	Opens the original input image.
Remove small objects	Removes small objects. A small object is defined by the number of erosions (specified in **Iterations**) needed to remove the object.
Remove large objects	Removes large objects. A large object is defined by the number of erosions (specified in **Iterations**) that do not remove the object.
Remove border objects	Eliminates particles that touch the borders of an image.
Fill holes	Fills holes found in a particle. Holes are filled with a pixel value of 1.
Convex Hull	Computes the convex hull of objects.
Skeleton	Applies a succession of thinnings until the width of each object equals one pixel. The Skeleton L function uses an L-shaped structuring element. The Skeleton M function extracts the skeleton with more dendrites. The Skiz function behaves like the Skeleton M function but it affects background regions, rather than object regions.
Separate objects	Breaks narrow isthmuses and separates objects that touch.
Label objects	Produces a labeled image using grayscale values equal to the number of objects in the image plus the grayscale 0 used in the background area.
Distance	Assigns to each pixel a grayscale value equal to the shortest distance to the border of the object, which might be to a hole within the object.

Danielsson	Creates a distance map similar to the Distance function. Danielsson uses a more accurate algorithm than Distance.
Segment image	Partitions an image into segments, each centered on an object, so that the segments do not overlap and leave empty zones.

- **Mode (Skeleton)**—Specifies the type of Skeleton function you want to perform. The Skeleton L function uses an L-shaped structuring element. The Skeleton M function extracts the skeleton with more dendrites. The Skiz function behaves like the Skeleton L function but it affects background regions, rather than object regions.
- **Structuring Element**

  2D array used as a binary mask to define the neighbors of a pixel. You can modify the structuring element by clicking its cells. If a cell is black, it has a value of 1. If a cell is white (empty), it has a value of 0. If cell is black, the corresponding pixel is considered to be a neighbor and its value is used during the morphology operation.
- **Size**—Size of the structuring element. Valid values include 3 x 3, 5 x 5, and 7 x 7.
- **Iterations**—Specifies the number of 3 x 3 erosions to apply to the image.
- **Connectivity 4/8**—Specifies how the algorithm determines if an adjacent pixel is the same or a different particle.

  Connectivity-8. All adjacent pixels are considered neighbors.

  Connectivity-4. Only pixels adjacent in the horizontal and vertical directions are considered neighbors.
- **Square/Hexagon**—Specifies if the pixel frame is treated as a square or hexagon during the transformation.

  Square pixel frame. Each pixel is surrounded by eight neighbors. The vertical and horizontal neighbors have a distance *d* from the pixel. Diagonal pixels have a slightly greater distance because they are farther away from the central pixel.

Hexagon pixel frame. Each pixel is surrounded by six neighbors. Each neighbor is at an equal distance from the central pixel.

Related Topics

Computing the convex hull of objects

Creating a distance map

Filling holes

Labeling objects

Reducing objects to their skeletons

Removing large objects

Removing objects touching the border of an image

Removing small objects

Segmenting an image

Separating objects

Removing small objects

1. Click **Binary»Adv. Morphology** or click **Adv. Morphology** in the **Binary** tab of the Processing Functions palette.
2. Select **Remove small objects**.
3. Specify the number of 3 x 3 erosions that need to be applied to completely remove a small object in **Iterations**.
4. Click **OK** to remove small objects and add this step to the script.

Related Topics

Adv. Morphology (Binary»Adv. Morphology)

Computing the convex hull of objects

Creating a distance map

Filling holes

Labeling objects

Reducing objects to their skeletons

Removing large objects

Removing objects touching the border of an image

Segmenting an image

Separating objects

Removing large objects

1. Click **Binary»Adv. Morphology** or click **Adv. Morphology** in the **Binary** tab of the Processing Functions palette.
2. Select **Remove large objects**.
3. Specify the number of 3 x 3 erosions to apply in **Iterations**. The particles that remain after the erosion operations are removed from the image and the rest of the particles are restored.
4. Click **OK** to remove large objects and add this step to the script.

Related Topics

Adv. Morphology (Binary»Adv. Morphology)

Computing the convex hull of objects

Creating a distance map

Filling holes

Labeling objects

Reducing objects to their skeletons

Removing objects touching the border of an image

Removing small objects

Segmenting an image

Separating objects

Removing objects touching the border of an image

1. Click **Binary»Adv. Morphology** or click **Adv. Morphology** in the **Binary** tab of the Processing Functions palette.
2. Select **Remove border objects**.
3. Click **OK** to remove all objects touching the border of the object and add this step to the script.

Related Topics

Adv. Morphology (Binary»Adv. Morphology)

Computing the convex hull of objects

Creating a distance map

Filling holes

Labeling objects

Reducing objects to their skeletons

Removing large objects

Removing small objects

Segmenting an image

Separating objects

Filling holes

1. Click **Binary»Adv. Morphology** or click **Adv. Morphology** in the **Binary** tab of the Processing Functions palette.
2. Select **Fill holes**.
3. Click **OK** to fill the holes and add this step to the script.

Related Topics

Adv. Morphology (Binary»Adv. Morphology)

Computing the convex hull of objects

Creating a distance map

Filling gaps and holes (close) in binary images

Filling gaps and holes (close) in grayscale images

Filling holes and smoothing right angles

Labeling objects

Reducing objects to their skeletons

Removing large objects

Removing objects touching the border of an image

Removing small objects

Segmenting an image

Separating objects

Computing the convex hull of objects

1. Click **Binary»Adv. Morphology** or click **Adv. Morphology** in the **Binary** tab of the Processing Functions palette.
2. Select **Convex**.
3. Click **OK** to close objects and add this step to the script.

Note This function is computer-intensive and can take several seconds.

Tip Use this function to close particles so you can make measurements even when the contour of the object is irregular.

Related Topics

Adv. Morphology (Binary»Adv. Morphology)

Creating a distance map

Filling holes

Labeling objects

Reducing objects to their skeletons

Removing large objects

Removing objects touching the border of an image

Removing small objects

Segmenting an image

Separating objects

Reducing objects to their skeletons

1. Click **Binary»Adv. Morphology** or click **Adv. Morphology** in the **Binary** tab of the Processing Functions palette.
2. Select **Skeleton**.
3. Select the type of skeleton you want to see.
   - **Skeleton L**—Uses an L-shaped structuring element.
   - **Skeleton M**—Extracts the skeleton with more dendrites.
   - **Skiz**—Uses an L-shaped structuring element that affects background regions, extracting the watershed line in an image.
4. Click **OK** to add this step to the script.

Note This function is computer-intensive and can take several seconds.

Related Topics

Adv. Morphology (Binary»Adv. Morphology)

Computing the convex hull of objects

Creating a distance map

Filling holes

Labeling objects

Removing large objects

Removing objects touching the border of an image

Removing small objects

Segmenting an image

Separating objects

Separating objects

1. Select **Binary»Adv. Morphology** or select **Adv. Morphology** in the **Binary** tab of the Processing Functions palette.
2. Select **Separate objects**.
3. Set the **Size** of the **Structuring Element**.
4. Click the cells of the structuring element that you want to have a value of 0.
5. Enter the number of erosions that you want the function to apply to the image in **Iterations**.
6. Click **OK** to separate objects and add this step to the script.

Notes

- The structuring element is a 2D array used as a binary mask. You can modify the structuring element by clicking its cells. If a cell is black, it has a value of 1. If a cell is white (empty), it has a value of 0.
- This function is computer-intensive and can take several seconds to process.

Tip Experiment with different coefficients and the **Kernel Size** to get the results you want.

Related Topics

Adv. Morphology (Binary»Adv. Morphology)

Computing the convex hull of objects

Creating a distance map

Filling holes

Labeling objects

Reducing objects to their skeletons

Removing large objects

Removing objects touching the border of an image

Removing small objects

Segmenting an image

Creating a distance map

1. Click **Binary»Adv. Morphology** or click **Adv. Morphology** in the **Binary** tab of the Processing Functions palette.
2. Select **Distance** or **Danielsson**.
3. Click **OK** to add this step to the script.

Note This function assigns, to each pixel, a grayscale value equal to the shortest distance to the border of the object. The border might be to a hole within the object. **Danielsson** uses a more accurate algorithm than **Distance**.

Related Topics

Adv. Morphology (Binary»Adv. Morphology)

Computing the convex hull of objects

Filling holes

Labeling objects

Reducing objects to their skeletons

Removing large objects

Removing objects touching the border of an image

Removing small objects

Segmenting an image

Separating objects

Labeling objects

1. Click **Binary»Adv. Morphology** or click **Adv. Morphology** in the **Binary** tab of the Processing Functions palette.
2. Select **Label objects**.
3. Click **OK** to add this step to the script.

Tip Labeling is especially useful when viewing images containing many objects that are not clearly separated.

Related Topics

Adv. Morphology (Binary»Adv. Morphology)

Computing the convex hull of objects

Creating a distance map

Filling holes

Reducing objects to their skeletons

Removing large objects

Removing objects touching the border of an image

Removing small objects

Segmenting an image

Separating objects

Segmenting an image

1. Click **Binary»Adv. Morphology** or click **Adv. Morphology** in the **Binary** tab of the Processing Functions palette.
2. Select **Segment image**.
3. Click **OK** to add this step to the script.

Notes

- This function partitions an image into segments, each centered on an object, so that the segments do not overlap and leave empty zones.
- This function is computer-intensive and can take several seconds.

Related Topics

Adv. Morphology (Binary»Adv. Morphology)

Computing the convex hull of objects

Creating a distance map

Filling holes

Labeling objects

Reducing objects to their skeletons

Removing large objects

Removing objects touching the border of an image

Removing small objects

Separating objects

Particle Filter (Binary»Particle Filter)

Removes or keeps particles in an image as specified by the filter criteria.

- **Filter Criteria**—List of measurements that you use as criteria to keep or remove particles from the image.
- **Parameter Range**

Minimum value	Specifies the minimum value of the parameter range.
Maximum value	Specifies the maximum value of the parameter range.
Exclude Interval	Specifies the range to filter. When this option is deselected, the selected range is [Minimum value, Maximum value]. When this option is selected, the selected range is [– ¥ , Minimum value] [Maximum value, + ¥]
Unit	Specifies the unit of the minimum and maximum value. The unit is either pixels or real-world values.

- **Current Parameter**—The minimum value, maximum value, and mean value for the selected filter criteria, which is calculated using all of the particles in the image.
- **Action**

Remove	Removes particles that do not fit in the range bounded by the minimal and maximal values for the selected measurement.
Keep	Keeps particles that fit in the range bounded by the minimal and maximal values for the selected measurement.

- **Connectivity 4/8**—Defines which of the surrounding pixels for any given pixel constitute its neighborhood.

▦	Connectivity-8. All adjacent pixels are considered neighbors.
⊞	Connectivity-4. Only pixels adjacent in the horizontal and vertical directions are considered neighbors.

Related Topics

Filtering objects based on particle measurements

Filtering objects based on particle measurements

1. Click **Binary»Particle Filter** or click **Particle Filter** in the **Binary** tab of the Processing Functions palette.
2. Double-click each object measurement in the **Particle Filter** list that you want the particle filter to include as criteria.
3. Enter the parameter range for each filter criteria.
4. Select the action you want to perform. **Remove** removes particles that do not fit in the range bounded by the minimum and maximum values for the selected measurement. **Keep** keeps particles that fit in the range bounded by the minimum and maximum values for the selected measurement.
5. Click **OK** to filter the image and add this step to the script.

Tip If you want the filter to be applied outside of the specified parameter range, check the **Exclude Interval** option.

Related Topics

Filter Criteria

Particle Filter (Binary»Particle Filter)

Threshold (Grayscale»Threshold)

Segments pixels in grayscale images. The Manual Threshold operation enables you to select ranges of grayscale pixel values. Local threshold operations select pixels using a locally adaptive thresholding algorithm. Use local thresholds in applications whose images exhibit non-uniform lighting changes that may result from a strong illumination gradient or shadows. Automatic threshold operations select threshold ranges for you. Use automatic thresholds when you expect uniform lighting changes from image to image.

- **Look For**

  Bright Objects Isolates pixels whose intensity values range from a user-specified value to 255.

  Dark Objects Isolates pixels whose intensity values range from 0 to a user-specified value.

  Gray Objects Isolates pixels whose intensity values lie within a user-specified range.

- **Threshold Type**

  Image Source Opens the original input image.

  Manual Threshold Applies a threshold to the image based on the Minimum and Maximum threshold values that you enter. All pixels not contained between the Minimum and Maximum values are set to 0. All pixels that fall in the range are replaced by 1.

  Local Threshold: Niblack Calculates a threshold value for each pixel based on the statistics of surrounding pixels. This algorithm compensates for high lighting variations.

  Local Threshold: Background Correction Performs a background correction to eliminate non-uniform lighting effects and then performs thresholding using the interclass variance thresholding algorithm.

  Auto Threshold: Clustering Applies a threshold to an image based on a statistical technique called clustering.

Auto Threshold: Entropy
Applies a threshold to an image based on an image analysis technique called entropy.

Auto Threshold: Metric
Applies a threshold to an image by calculating the optimal threshold, which depends on the surfaces representing the initial grayscale, using the metric technique.

Auto Threshold: Moments
Applies a threshold to an image by using a statistical tool called moments, which recalculates a theoretical binary image.

Auto Threshold: Inter Variance
Applies a threshold to an image based on a classical statistical technique called interclass variance.

- **Threshold Range**—Configurable only with a Manual Threshold.

  Minimum
  Lowest grayscale pixel value included in the manual threshold range. You can adjust the black cursor on the histogram to set the lower bound for this range when you are looking for bright or gray objects.

  Maximum
  Highest grayscale pixel value included in the manual threshold range. You can adjust the white cursor on the histogram to set the upper bound for this range when you are looking for dark or gray objects.

- **Histogram**—Displays the total number of pixels in each grayscale value. From the histogram, you can discern if the image contains distinct regions of a certain grayscale value, and you can select grayscale pixel regions of the image by sliding the white and black cursors.

  Tips

  - You can display the histogram of a portion of an image by drawing an ROI around the area of interest with one of the tools from the toolbar. The local histogram gives you valuable information about the threshold range you can select to keep or reject certain parts of the image.

    - Thresholding an image produces a binary images with pixel values of 0 and 1. You must use a binary color palette to display binary images.

- **Kernel Size**—Size of the neighborhood around each pixel used to compute the mean value. The Kernel Size is typically the size of the object you want to isolate with the threshold. This control is configurable only with a local threshold.

  Tip You can get the approximate size of an object in your image by drawing an ROI around the object. ROI Size displays the size of the last ROI drawn. Click << to apply the ROI Size to the Kernel Size.

- **Deviation Factor**—Determines the sensitivity of the algorithm. Values range from 0 to 1, with 0 being the most sensitive to noise. The lower the Deviation Factor, the closer the pixel value must be to the mean value to be selected as part of a particle. This control is configurable only with the Local Threshold: Niblack Threshold Type.

Related Topics

Color Threshold (Color»Color Threshold)

Thresholding a grayscale image

Color Threshold (Color»Color Threshold)

Applies a threshold to the three planes of a color image and places the result into an 8-bit image.

- **Color Model**—Color encoding scheme: RGB, HSL, HSV, or HSI.
- **Preview Color**—Color used to highlight the pixels selected by the threshold **Min** and **Max** values. Default value is red.
- **Histogram**

  Linear — Displays the histograms in a linear scale.

  Logarithmic — Displays the histograms in a logarithmic scale. Use the logarithmic scale if the linear values are too small to view using a linear scale.
- **Red/Hue**—Thresholding range for the red or hue plane (depending on **Color Model**). Any pixel values not included in this range are reset to zero in the destination image. You can modify the **Min** and **Max** values by clicking and dragging on the blue and red markers on the histogram.
- **Green/Saturation**—Thresholding range for the green or saturation plane (depending on **Color Model**). Any pixel values not included in this range are reset to zero in the destination image. You can modify the **Min** and **Max** values by clicking and dragging on the blue and red markers on the histogram.
- **Blue/Luminance/Value/Intensity**—Thresholding range for the blue, luminance, value, or intensity plane (depending on **Color Model**). Any pixel values not included in this range are reset to zero in the destination image. You can modify the **Min** and **Max** values by clicking and dragging on the blue and red markers on the histogram.

  Note You can use the Hue indicator to select all pixels with a certain hue when you set **Color Model** to HSL, HSV, or HSI. Select a specific hue that you want to isolate in the image using the ROI tools. The histograms display the pixels that you need to select in order to isolate all of pixels with that hue. Drag the blue and red markers on the histograms until all pixels with that hue are selected.

- **Min**—Lowest pixel value included in the manual threshold range. For manual thresholds, you can adjust the blue cursor on the histogram to set the lower bound for this range.
- **Max**—Highest pixel value included in the manual threshold range. For manual thresholds, you can adjust the red cursor on the histogram to set the upper bound for this range.
- **In/Out Range (Valid for HSL, HSV, or HSI)**

  □ **In—**Hue is included in the threshold range. Threshold range: [Min, Max]

  Out—Hue is not included in the threshold range. Threshold range: [0, Min] [Max, 255]

Tip You can display the histogram of a portion of an image by drawing an ROI around the area of interest. The local histogram gives you valuable information about the threshold range you can select to keep or reject certain parts of the image.

Related Topics

Grayscale Threshold (Grayscale»Threshold)

Thresholding an image

Thresholding an image

You can threshold grayscale and color images.

Related Topics

Color Threshold (Color»Color Threshold)

Grayscale Threshold (Grayscale»Threshold)

Watershed Segmentation (Grayscale»Watershed Segmentation)

Performs segmentation on images using the watershed transform. Refer to the *NI Vision Concepts Manual* for information about the watershed transform.

- **Number of Zones**—Displays the total number of zones detected in the image.
- **Connectivity 4/8**—Defines which of the surrounding pixels for any given pixel constitute its neighborhood.
  - **Connectivity-8**—All adjacent pixels are considered neighbors.
  - **Connectivity-4**—Only pixels adjacent in the horizontal and vertical directions are considered neighbors.
- **Display**—Controls how Vision Assistant displays the image after watershed segmentation.
  - **Labeled Binary Output**—Displays the labeled binary image returned by the watershed algorithm.
  - **Image Source with Separation Overlay**—Overlays the watershed lines in red on top of the input image. This display is helpful to see if the zones found are correct. This is for viewing purposes only and does not change the output of the step.

  Note The Display options control how the results are displayed in the Processing window when the Setup window is open. However, the output image of the step is always a labeled binary image.

Related Topics

Segmenting an image

Segmenting an image using a watershed transform

Note The watershed function works only with 8-bit and 16-bit images.

The following steps illustrate one method of segmenting objects.

1. Threshold—(**Grayscale»Threshold**) Threshold the image to isolate the objects you want to segment from the rest of the image.
2. Remove Noise—(**Binary»Advanced Morphology** or **Binary»Particle Filter**) Remove extraneous information from the binary image using various morphology operations. For example, you can fill in holes with the Advanced Morphology option Fill holes or remove noise with the Particle Filter function.
3. Create Distance Map—(**Binary»Advanced Morphology**) Use the Danielsson option to transform the binary image into a grayscale image in which the center of each object represents a local maximum in the image. If you were to look at the topographical surface of the image after applying the distance map, each object would appear as a smooth, gradual peak.
4. Apply Watershed Transform—(**Grayscale»Watershed Segmentation**) Apply the watershed transform to segment the image into local zones. A zone is determined by the watershed line that separates the influence zone of each peak.

Refer to the *NI Vision Concepts Manual* for more information about thresholding, morphology, distance transforms, or watershed transforms.

Related Topics

Threshold (Grayscale»Threshold)

Morphology (Binary»Advanced Morphology)

Particle Filter (Binary»Particle Filter

Watershed Segmentation FAQs

How can I label more than 255 zones?

The watershed transform accepts 8-bit and 16-bit images. The maximum number of labels in an 8-bit image is 255. Zones over 255 are saturated at 255. The maximum number of labels in a 16-bit grayscale image is 32,766. If an 8-bit image requires more than 255 labels, cast the image to a 16-bit image before using the Watershed Segmentation step.

If 8-bit images can label up to only 255 zones, is the Number of Zones result still accurate if more than 255 zones are detected?

Watershed Segmentation returns the correct number of zones found even if the number of zones is greater than the maximum number of labels for a particular bit depth.

Can I use the output of the watershed transform to separate touching objects?

The Watershed Segmentation algorithm partitions an image based on the topographic surface of the image. The image is separated into non-overlapping segments. Each segment contains a unique particle. Complete these steps to separate touching particles by masking the segmentation lines.

1. Store a copy of the binary image in Buffer #1 with the Image Buffer step (**Image»Image Buffer**).
2. Use the Danielsson option (**Binary»Adv. Morphology»Danielsson**) to assign to each pixel a grayscale value equal to the shortest distance to the border of the object.
3. Use the Watershed Segmentation step (**Grayscale»Watershed Segmentation**) to partition the image.
4. Store the watershed image in Buffer #2 with the Image Buffer step.
5. Retrieve the thresholded image from Buffer #1 using the Image Buffer step.
6. Mask the thresholded image with the watershed image by using the Mask Buffer step (**Grayscale»Operators»Mask**).

Particle

Use any of the following measurements or a combination of the following measurements to keep or remove particles from the image.

- **Center of Mass X and Center of Mass Y**—X- and Y-coordinates of the particle center of mass.
- **First Pixel X and First Pixel Y**—X- and Y-coordinates of the first pixel of the particle.
- **Bounding Rect (Left, Right, Top, and Bottom)**
  - X-coordinate of the leftmost point in the particle
  - Y-coordinate of highest point in the particle
  - X-coordinate of the rightmost point in the particle
  - Y-coordinate of the lowest point in the particle
- **Max Feret Diameter Start X and Max Feret Diameter Start Y**—X- and Y-coordinates of the particle center the Max Feret Diameter Start.
- **Max Feret Diameter End X and Max Feret Diameter End Y**—X- and Y-coordinates of the particle center the Max Feret Diameter End.
- **Max Horiz. Segment Length (Left, Right, and Row)**
  - X-coordinate of the leftmost pixel in the Max Horiz. Segment. Always given as a pixel measurement.
  - X-coordinate of the rightmost pixel in the Max Horiz. Segment. Always given as a pixel measurement.
  - Y-coordinate for all of the pixels in the Max Horiz. Segment. Always given as a pixel measurement.
- **Bounding Rect (Width, Height, and Diagonal)**
  - Distance between Bounding Rect Left and Bounding Rect Right
  - Distance between Bounding Rect Top and Bounding Rect Bottom
  - Distance between opposite corners of the Bounding Rect
- **Perimeter**—Length of the outer boundary of the particle. Because the boundary is comprised of discrete pixels, Vision Assistant

subsamples the boundary points to approximate a smoother, more accurate perimeter.

- **Convex Hull Perimeter**—Perimeter of the Convex Hull.
- **Hole's Perimeter**—Sum of the perimeters of each hole in the particle.
- **Max Feret Diameter**—Distance between the Max Feret Diameter Start and the Max Feret Diameter End.
- **Equivalent Ellipse (Major Axis, Minor Axis, and Minor Axis (Feret))**
    - Length of the major axis of the Equivalent Ellipse
    - Length of the minor axis of the Equivalent Ellipse
    - Length of the minor axis of the ellipse with the same area as the particle, and Major Axis equal in length to the Max Feret Diameter
- **Equivalent Rect (Long Side, Short Side, Diagonal, and Short Side (Feret))**
    - Longest side of the Equivalent Rect
    - Shortest side of the Equivalent Rect
    - Distance between opposite corners of the Equivalent Rect
    - Shortest side of the rectangle with the same area as the particle, and longest side equal in length to the Max Feret Diameter
- **Average Horiz. Segment Length**—Average length of a horizontal segment in the particle. Sum of the horizontal segments that do not superimpose any other horizontal segment. Always given as a pixel measurement.
- **Average Vert. Segment Length**—Average length of a vertical segment in the particle. Sum of the vertical segments that do not superimpose any other vertical segment. Always given as a pixel measurement.
- **Hydraulic Radius**—Particle area divided by its perimeter.
- **Waddel Disk Diameter**—Diameter of a disk with the same area as the particle.
- **Area**—Area of the particle.

- **Holes' Area**—Sum of the areas of each hole in the particle.
- **Particle & Holes' Area**—Area of a particle that completely covers the image.
- **Convex Hull Area**—Area of the Convex Hull of the particle.
- **Image Area**—Area of the image.
- **Number of Holes**—Number of holes in the particle.
  Vision Assistant can detect holes inside a particle as small as one pixel.
- **Number of Horiz. Segments**—Number of horizontal segments in the particle. always given as pixel measurements.
- **Number of Vert. Segments**—Number of horizontal segments in the particle. Always given as pixel measurements.
- **Orientation**—Angle of the line passing through the particle center of mass with the lowest moment of inertia.
- **Max Feret Diameter Orientation**—Angle of the Max Feret Diameter.
- **%Area/Image Area**—Percentage of the particle's Area covering the Image Area.
- **%Area/(Particle & Holes' Area)**—Percentage of the particle's Area in relation to its Particle & Holes' Area.
- **Ratio of Equivalent Ellipse Axes**—Equivalent Ellipse Major Axis divided by Equivalent Ellipse Minor Axis.
- **Ratio of Equivalent Rect Sides**—Equivalent Rect Long Side divided by Equivalent Rect Short Side.
- **Elongation Factor**—Max Feret Diameter divided by Equivalent Rect Short Side (Feret). The more elongated the shape of a particle, the higher its elongation factor.
- **Compactness Factor**—Area divided by the product of Bounding Rect Width and Bounding Rect Height. The compactness factor belongs to the interval [0, 1]. The closer the shape of a particle is to a rectangle, the closer to 1 the compactness factor.
- **Heywood Circularity Factor**—Perimeter divided by the circumference of a circle with the same area. The closer the shape of a particle is to a disk, the closer the Heywood circularity factor to 1.

- **Type Factor**—Factor relating area to moment of inertia.
- **Angle**—Degrees of rotation counterclockwise from the x-axis. Range: [0 °, 180 °).
- **Sum...**—Moments of various orders relative to the x- and y-axes.
- **Moment of Inertia...**—Moments about the particle Center of Mass. Gives a representation of the distribution of the pixels in a particle with respect to its center of gravity. Shift invariant.
- **Norm. Moment of Inertia...**—Moment of Inertia normalized with regard to the particle Area. Shift and scale invariant.
- **Hu Moment...**—Moments derived from the Norm. Moment of Inertia measurements. Shift, scale, and rotation invariant.

Thresholding a grayscale image

1. Click **Grayscale»Threshold** or click **Threshold** in the **Grayscale** tab of the Processing Functions palette.
2. Select whether you want the function to look for bright, dark, or gray objects.
3. Select the type of threshold you want to perform.

Note You cannot adjust the threshold levels of the local and automatic thresholds. If none of the local or automatic thresholding options work for the image, select **Manual Threshold** and adjust the pointers on the histogram until all of the particles of interest are highlighted in red.

3. Click **OK** to threshold the image and add this step to the script.

Tips

- If you have trouble finding a manual threshold range that selects all of the pixels in the range, draw an ROI around the portion of the image that you want to keep. The histogram displays the intensities of the ROI only. Use the minimum and maximum values for the threshold.

Related Topics

Threshold (Grayscale»Threshold)

Thresholding an image

1. Click **Grayscale»Threshold** or click **Threshold** in the **Grayscale** tab of the Processing Functions palette.
2. Select the type of threshold you want to perform.

Note You cannot adjust the threshold levels of the automatic thresholds. If none of the automatic thresholds work for the image, select **Manual Threshold** and adjust the pointers on the histogram until all of the particles of interest are highlighted in red.

3. Click **OK** to threshold the image and add this step to the script.

Tips

- If you have trouble finding a threshold range that selects all of the pixels in the range, draw an ROI around the portion of the image that you want to keep. The histogram displays the intensities of the ROI only. Use the minimum and maximum values for the threshold.
- You can display the histogram in either linear or logarithmic form.

Related Topics

Grayscale Threshold (Grayscale»Threshold)

Thresholding an image

1. Click **Color»Color Threshold** or click **Color Threshold** in the **Color** tab of the Processing Functions palette.
2. Select the color model (RGB, HSL, HSV, or HSI) you want to use to process the image.
3. Adjust the blue and red lines on the **Red (Hue)**, **Green (Saturation)**, and **Blue (Luminance/Value/Intensity)** histograms until all of the particles of interest are highlighted with the color specified in **Preview Color**.
4. Click **OK** to threshold the image and add this step to the script.

Tips

- If you have trouble finding a threshold range that selects all of the pixels in the range, draw an ROI around the portion of the image that you want to keep. The histogram displays the intensities of the ROI only. Use the minimum and maximum values for the threshold.
- You can display the histogram in either linear or logarithmic form.

Related Topics

Color Threshold (Color»Color Threshold)