Filter

- Smooth Data and Reduce Noise
- Perform Edge Detection
- Perform Linear Feature Detection
- Enhance Remote Sensing Imagery

Examples
- Data/Image Smoothing
- Despeckling and Noise Reduction
- Edge Detection
- Remote Sensing Image Enhancement
- Shaded Relief
- Pseudo Shaded Relief
- Generate Contours From a DEM

Dialog Box
- Using the operation dialog box interface
- Using the dialog box interface to create or edit scripts

Map
Use the Map drop-down list to specify a map layer to be filtered. There are three main purposes to filtering map layers: To improve interpretability of image data; to aid in automated feature extraction; and to remove and/or...
reduce data “noise”. The **Filter** operation is designed for use with continuous data sets such as DEMs (Digital Elevation Models), airphotos, and satellite imagery.

**Kernel**
Each of the filters uses a 3 cell by 3 cell roving, convolving kernel to assign values to the resultant map layer. You can create and specify your own convolving kernel to be used in place of the default kernel. Your kernel can be any size as long as it is square and is composed of an odd number of rows and columns.

**How Do I Create a Convolving Kernel for the Filter Operation?**

The **Sobel** filter and the **Difference** filters are the only filters for which you cannot define a new kernel.

**Note:** you should have prior knowledge of the workings of remote sensing image filtering if you plan to create your own convolving kernels. The default kernel supplied with each filter is the standard kernel for that filter.

Use the **Kernel** drop-down list to specify a kernel that you have created. Specify **None** to use the default kernel.

**Filter Low Pass**
The image filtering term “pass” means to emphasize. The **Low Pass** filter emphasizes low frequency features and filters out, or suppresses, high frequency changes in data values. You can use the **Low Pass** filter to smooth data or despeckle images.

The **Low Pass** filter is a convolving filter that reduces local deviations through averaging. The default kernel is 3 cells by 3 cells. You can specify your own kernel to increase the distance (such as 5x5, 7x7, or 9x9) over which values will be averaged. The **Low Pass** filter is also referred to as a “mean” filter because it is used to generalize or smooth data by calculating the mean of the neighbours of the cell being assigned a new value. The formula for calculating the mean is:

\[
\bar{x} = \frac{\sum (x_i \times w_i)}{n}
\]

Where:
- \( x_i \) = cells in the kernel
\[ w_i = \text{weight assigned to cell by the kernel} \]
\[ n = \text{number of cells in the kernel} \]

The default kernel assigns a weighting of 1 to each cell in the kernel. The cells in the kernel are added together and the mean is calculated. Here is the default kernel for the **Low Pass** filter:

![Low Pass Default Kernel](image)

If you supply a kernel, the values of the kernel will be treated as multipliers \((w_i)\) for the cell values \((x_i)\) before calculating the mean \((x)\).

The colour sequence that you apply to a filtered map layer greatly affects the appearance of the results.

- How Do I Apply a Colour Sequence to Enhance Filter Results?
- How Do I Create a Convolving Kernel for the Filter Operation?

**Filter Chavez**

This modifier specifies a high pass image enhancement filter. The image filtering term “pass” means to emphasize. The **Chavez** filter emphasizes high frequency features and filters out, or suppresses, low frequency changes in data values. You can use the **Chavez** filter to extract or emphasize abrupt changes in data values from a cell to its neighbours. High pass filters enhance fine detail, edges, and linear features; however, details in flat areas are lost.

If the image contains noise, this edge detection filter will emphasize the noise. The effects of noise should be removed by a low pass filter before running edge enhancement filters.

The default kernel supplies a matrix of multipliers to the filter. If you create your own kernel, the values of that kernel will be treated as multipliers for
the corresponding cell values. Each cell in the default kernel carries a weight of 1:

\[
\begin{array}{ccc}
-1 & -1 & -1 \\
-1 & 17 & -1 \\
-1 & -1 & -1 \\
\end{array}
\]

Chavez Default Kernel

The new cell value is calculated as 1/9 times the average of the weighted values in the window.

\[
x = \frac{1}{9} \times \left( \frac{\sum(x_i \times w_i)}{n} \right)
\]

Where:
- \(x_i\) = cells in the kernel
- \(w_i\) = weight assigned to cell by the kernel
- \(n\) = number of cells in the kernel

The colour sequence that you apply to a filtered map layer greatly affects the appearance of the results.

How Do I Apply a Colour Sequence to Enhance Filter Results?
How Do I Create a Convolving Kernel for the Filter Operation?

**Filter Laplacian Type 1**
The Laplacian Type 1 filter is a standard edge enhancement filter. It is designed to emphasize high frequency changes in the data through edge detection. The Laplacian Type 1 filter is a high pass image enhancement filter. The image filtering term “pass” means to emphasize. The Laplacian Type 1 filter emphasizes high frequency features and filters out, or suppresses, low frequency changes in data values. Use the Laplacian Type 1 filter to extract or emphasize abrupt changes in data values from a cell to its neighbours. High pass filters enhance fine detail, edges, and linear features; however, details in flat areas are lost.

If the image contains noise, this edge detection filter will emphasize the noise. The effects of noise should be removed by a low pass filter before running edge enhancement filters.
The default kernel supplies a matrix of multipliers to the filter. If you create your own kernel the values of that kernel will be treated as multipliers for the corresponding cell values. The default kernel for this filter is:

\[
\begin{array}{ccc}
0 & 1 & 0 \\
1 & -4 & 1 \\
0 & 1 & 0 \\
\end{array}
\]

The new cell value is calculated as the sum of all cell values multiplied by the weights in the kernel:

\[ x = \sum (x_i \times w_i) \]

Where:
- \( x_i \) = cells in the kernel
- \( w_i \) = weight assigned to cell by the kernel

The colour sequence that you apply to a filtered map layer greatly affects the appearance of the results.

- How Do I Apply a Colour Sequence to Enhance Filter Results?
- How Do I Create a Convolving Kernel for the Filter Operation?

**Filter Laplacian Type 2**

This modifier specifies a high pass image enhancement filter. The image filtering term “pass” means to emphasize. The Laplacian Type 2 filter emphasizes high frequency features and filters out, or suppresses, low frequency changes in data values. You can use the Laplacian Type 2 filter to extract or emphasize abrupt changes in data values from a cell to its neighbours. High pass filters enhance fine detail, edges, and linear features; however, details in flat areas are lost.

If the image contains noise, this edge detection filter will emphasize the noise. The effects of noise should be removed by a low pass filter before running edge enhancement filters.
The default kernel supplies a matrix of multipliers to the filter. If you create your own kernel the values of that kernel will be treated as multipliers for the corresponding cell values. The default kernel for this filter is:

```
-1 -1 -1
-1  8 -1
-1 -1 -1
```

Laplacian 2
Default Kernel

The new cell value is calculated as the sum of all cells multiplied by the kernel weights:

\[ x = \sum (x_i \times w_i) \]

Where:
- \( x_i \) = cells in the kernel
- \( w_i \) = weight assigned to cell by the kernel

The **Laplacian Type 2** filter is an alternative to the Chavez filter; it gives greater emphasis to abrupt changes in data values from a cell to its neighbours.

The colour sequence that you apply to a filtered map layer greatly affects the appearance of the results.

- How Do I Apply a Colour Sequence to Enhance Filter Results?
- How Do I Create a Convolving Kernel for the Filter Operation?

**Filter Sobel**

The **Sobel** modifier specifies the use of the **Sobel** filter, a standard edge enhancement filter. The **Sobel** filter emphasizes the high frequency changes in the data through edge detection. The **Sobel** filter greatly exaggerates the effects of noise in remote sensing imagery. If possible, use this filter on imagery that has undergone preprocessing to despeckle or reduce noise.

The **Sobel** filter generates an image where sharp changes in data values are greatly emphasized. The **Sobel** filter is based on gradient discontinuity. Gradient is a vector whose components measure how rapidly pixel values
are changing with distance in the x and y directions; therefore, the Sobel filter is directional; it is sensitive to both horizontal and vertical edges.

You cannot specify a custom kernel map for this modifier. The Sobel filter uses a 3 cell by 3 cell window as follows:

![Sobel Built-in Kernel](image)

Cell values are calculated using the following formulae:

\[
A = (V_3 + 2V_6 + V_9) - (V_1 + 2V_4 + V_7)
\]

\[
B = (V_1 + 2V_2 + V_3) - (V_7 + 2V_8 + V_9)
\]

\[
X = \sqrt{A^2 + B^2}
\]

The colour sequence that you apply to a filtered map layer greatly affects the appearance of the results.

**How Do I Apply a Colour Sequence to Enhance Filter Results?**

**Filter Horizontal Difference**

The **Horizontal Difference** filter is a uni-directional edge detection filter. Values are assigned based on the difference in value from one cell to the cell directly below. If cell values are the same, then a value of “0” is assigned. If there is a positive difference, a value of “1” will be assigned. If there is a negative difference, a value of “-1” is assigned.

If the **Horizontal Difference** filter is applied to remote sensing imagery, surface roughness is emphasized. Linear features with horizontal trends are highlighted. When applied to DEMs (Digital Elevation Models) or sliced elevation data, this has the effect of creating a “contoured” pseudo shaded relief map layer with lighting either from the top or the bottom of the map layer depending on the colour sequence that you apply.
The optimal colour scheme for map layers generated by the **Horizontal Difference** filter is “-1” = 100% Red, 100% Green, 100% Blue; “0” = 50% Red, 50% Green, 50% Blue; and “1” = 0% Red, 0% Green, 0% Blue. If this colour sequence produces a “reverse” image to your eye (i.e., hollows appear to be hills and hills appear to be hollows) then try reversing this colour sequence.

**Filter Vertical Difference**
The **Vertical Difference** filter is a uni-directional edge detection filter. Values are assigned based on the difference in value from one cell to the cell directly to the right. If cell values are the same, then a value of “0” is assigned. If there is a positive difference, a value of “1” will be assigned. If there is a negative difference, a value of “-1” is assigned.

If the **Vertical Difference** filter is applied to remote sensing imagery, surface roughness is emphasized. Linear features with vertical trends are highlighted. When applied to DEMs (Digital Elevation Models) or sliced elevation data, this has the effect of creating a “contoured” pseudo shaded relief map layer with lighting either from the left or the right of the map layer depending on the colour sequence that you apply.

The optimal colour scheme for map layers generated by the **Vertical Difference** filter is “-1” = 100% Red, 100% Green, 100% Blue; “0” = 50% Red, 50% Green, 50% Blue; and “1” = 0% Red, 0% Green, 0% Blue. If this colour sequence produces a “reverse” image to your eye (i.e., hollows appear to be hills and hills appear to be hollows) then try reversing this colour sequence.

**Filter Diagonal Difference**
The **Diagonal Difference** filter is a uni-directional edge detection filter. Values are assigned based on the difference in value from one cell to the cell below and to the right. If these cell values are the same, then a value of “0” is assigned. If there is a positive difference, a value of “1” will be assigned. If there is a negative difference, a value of “-1” is assigned.

If the **Diagonal Difference** filter is applied to remote sensing imagery, surface roughness is emphasized. Linear features with diagonal trends are highlighted. When applied to DEMs (Digital Elevation Models) or sliced elevation data, this has the effect of creating a “contoured” pseudo shaded relief map layer with lighting either from the top left or the bottom right of the map layer depending on the colour sequence that you apply.

The optimal colour scheme for map layers generated by the **Diagonal Difference** filter is “-1” = 100% Red, 100% Green, 100% Blue; “0” = 50% Red, 50% Green, 50% Blue; and “1” = 0% Red, 0% Green, 0% Blue. If this colour sequence produces a “reverse” image to your eye (i.e., hollows
appear to be hills and hills appear to be hollows) then try reversing this colour sequence.

Syntax

- Syntax and type conventions
- Using the Script window interface
- Using the dialog box interface to create or edit scripts

Filter map [Kernel map]
   [LowPass | Chavez | Laplacian1 | Laplacian2 | Sobel | DiffHoriz | DiffVert | DiffDiag];

Filter map
Use the Filter statement to specify a map layer to be smoothed or enhanced. There are three main purposes to filtering map layers: To improve interpretability of image data; to aid in automated feature extraction; and to remove and/or reduce data “noise”. The filter operation is designed for use with continuous data sets such as DEMs (Digital Elevation Models), air photos, and satellite imagery.

Kernel map
Each of the filters uses a 3 cell by 3 cell roving, convolving kernel to assign values to the resultant map layer. You can create and specify your own convolving kernel to be used in place of the default kernel. Your kernel map can be any size as long as it is square and is composed of an odd number of rows and columns.

How Do I Create a Convolving Kernel for the Filter Operation?

The Sobel filter and the Difference filters are the only filters for which you cannot define a new kernel.

Note: you should have prior knowledge of the workings of remote sensing image filtering if you plan to create your own convolving kernels. The default kernel supplied with each filter is the standard kernel for that filter.

LowPass
The image filtering term “pass” means to emphasize. The LowPass filter emphasizes low frequency features and filters out, or suppresses, high frequency changes in data values. You can use the LowPass filter to smooth data or despeckle images.

The LowPass filter is a convolving filter that reduces local deviations by averaging. The default kernel is 3 cells by 3 cells. You can specify your own kernel to increase the distance (such as 5x5, 7x7, or 9x9) over which values
will be averaged. The **LowPass** filter is also referred to as a “mean” filter because it is used to generalize or smooth data by calculating the mean of the neighbours of the cell being assigned a new value. The formula for calculating the mean is:

\[
x = \frac{\sum (x_i \times w_i)}{n}
\]

Where:
- \(x_i\) = cells in the kernel
- \(w_i\) = weight assigned to cell by the kernel
- \(n\) = number of cells in the kernel

The default kernel assigns a weighting of 1 to each cell in the kernel. The cells in the kernel are added together and the mean is calculated. Here is the default kernel for the **LowPass** filter:

```
1 1 1
1 1 1
1 1 1
```

The colour sequence that you apply to a filtered map layer greatly affects the appearance of the results.

- **How Do I Apply a Colour Sequence to Enhance Filter Results?**
- **How Do I Create a Convolving Kernel for the Filter Operation?**

**Chavez**

This modifier specifies a high pass image enhancement filter. The image filtering term “pass” means to emphasize. The **Chavez** filter emphasizes high frequency features and filters out, or suppresses, low frequency changes in data values. You can use the **Chavez** filter to extract or emphasize abrupt changes in data values from a cell to its neighbours. High pass filters enhance fine detail, edges, and linear features; however, details in flat areas are lost.
The default kernel supplies a matrix of multipliers to the filter. If you create your own kernel, the values of that kernel will be treated as multipliers for the corresponding cell values. Each cell in the default kernel weights carries a weight of 1:

\[
\begin{pmatrix}
-1 & -1 & -1 \\
-1 & 17 & -1 \\
-1 & -1 & -1
\end{pmatrix}
\]

Chavez Default Kernel

The new cell value is calculated as 2x’s the centre value in the window, minus the average of the weighted values in the window.

\[
x = 2x_{centre} - \frac{\sum (x_i \times w_i)}{n}
\]

Where:
- \(x_i\) = cells in the kernel
- \(x_{centre}\) = the centre cell of the kernel
- \(w_i\) = weight assigned to cell by the kernel
- \(n\) = number of cells in the kernel

The colour sequence that you apply to a filtered map layer greatly affects the appearance of the results.

- How Do I Apply a Colour Sequence to Enhance Filter Results?
- How Do I Create a Convolving Kernel for the Filter Operation?

Laplacian1

The Laplacian1 filter is a standard edge enhancement filter. It is designed to emphasize high frequency changes in the data through edge detection. The Laplacian1 filter is a high pass image enhancement filter. The image filtering term “pass” means to emphasize. The Laplacian1 filter emphasizes high frequency features and filters out, or suppresses, low frequency changes in data values. Use the Laplacian1 filter to extract or emphasize abrupt changes in data values from a cell to its neighbours. High pass filters enhance fine detail, edges, and linear features; however, details in flat areas are lost.
The default kernel supplies a matrix of multipliers to the filter. If you create your own kernel, the values of that kernel will be treated as multipliers for the corresponding cell values. The default kernel for this filter is:

\[
\begin{bmatrix}
0 & 1 & 0 \\
1 & -4 & 1 \\
0 & 1 & 0 \\
\end{bmatrix}
\]

Laplacian 1
Default Kernel

The new cell value is calculated as the sum of the cell values multiplied by the weights in the kernel:

\[
x = \Sigma (x_i \times w_i)
\]

Where:
- \(x_i\) = cells in the kernel
- \(w_i\) = weight assigned to cell by the kernel

The colour sequence that you apply to a filtered map layer greatly affects the appearance of the results.

- How Do I Apply a Colour Sequence to Enhance Filter Results?
- How Do I Create a Convolving Kernel for the Filter Operation?

Laplacian2
The Laplacian2 modifier specifies a high pass image enhancement filter. The image filtering term “pass” means to emphasize. The Laplacian2 filter emphasizes high frequency features and filters out, or suppresses, low frequency changes in the data values from cell to cell. You can use the Laplacian2 filter to extract or emphasize abrupt changes in data values from a cell to its neighbours. High pass filters enhance fine detail, edges, and linear features; however, details in flat areas are lost.
The default kernel supplies a matrix of multipliers to the filter. If you create your own kernel, the values of that kernel will be treated as multipliers for the corresponding cell values. The default kernel for this filter is:

\[
\begin{pmatrix}
-1 & -1 & -1 \\
-1 & 8 & -1 \\
-1 & -1 & -1
\end{pmatrix}
\]

Laplacian 2
Default Kernel

The new cell value is calculated as the sum of all cells multiplied by the kernel weights:

\[ x = \sum (x_i \times w_i) \]

Where:
- \( x_i \) = cells in the kernel
- \( w_i \) = weight assigned to cell by the kernel

The Laplacian2 filter is an alternative to the Laplacian1 filter; it gives greater emphasis to abrupt changes in data values from a cell to its neighbours.

The colour sequence that you apply to a filtered map layer greatly affects the appearance of the results.

How Do I Apply a Colour Sequence to Enhance Filter Results?
How Do I Create a Convolving Kernel for the Filter Operation?

Sobel
The Sobel modifier specifies the use of the Sobel filter, a standard edge enhancement filter. The Sobel filter emphasizes the high frequency changes in the data through edge detection. The Sobel filter greatly exaggerates the effects of noise in remote sensing imagery. If possible, use this filter on imagery that has undergone preprocessing to despeckle or reduce noise.

The Sobel filter generates an image where sharp changes in data values are greatly emphasized. The Sobel filter is based on the gradient discontinuity. Gradient is a vector whose components measure how rapidly pixel values
are changing with distance in the x and y directions; therefore, the Sobel filter is directional; it is sensitive to both horizontal and vertical edges.

You cannot specify a custom kernel map for this modifier. The Sobel filter uses a 3 cell by 3 cell window as follows:

\[
\begin{array}{ccc}
V_1 & V_2 & V_3 \\
V_4 & X & V_6 \\
V_7 & V_8 & V_9 \\
\end{array}
\]

Sobel Built-in Kernel

Cell values are calculated using the following formula:

\[
A = (V_3 + 2V_6 + V_9) - (V_1 + 2V_4 + V_7)
\]

\[
B = (V_1 + 2V_2 + V_3) - (V_7 + 2V_8 + V_9)
\]

\[
X = \sqrt{A^2 + B^2}
\]

The colour sequence that you apply to a filtered map layer greatly affects the appearance of the results.

- How Do I Apply a Colour Sequence to Enhance Filter Results?
- How Do I Create a Convolving Kernel for the Filter Operation?

DiffHoriz

The DiffHoriz filter is a uni-directional edge detection filter. Values are assigned based on the difference in value from one cell to the cell directly below. If cell values are the same, then a value of “0” is assigned. If there is a positive difference, a value of “1” will be assigned. If there is a negative difference, a value of “-1” is assigned.

If the DiffHoriz filter is applied to remote sensing imagery, surface roughness is emphasized. Linear features with horizontal trends are highlighted. When applied to DEMs (Digital Elevation Models) or sliced elevation data, this has the effect of creating a “contoured” pseudo shaded
relief map layer with lighting either from the top or the bottom of the map layer depending on the colour sequence that you apply.

The optimal colour scheme for map layers generated by the **DiffHoriz** filter is “-1” = 100% Red, 100% Green, 100% Blue; “0” = 50% Red, 50% Green, 50% Blue; and “1” = 0% Red, 0% Green, 0% Blue. If this colour sequence produces a “reverse” image to your eye (i.e., hollows appear to be hills and hills appear to be hollows) then try reversing this colour sequence.

**DiffVert**
The **DiffVert** filter is a uni-directional edge detection filter. Values are assigned based on the difference in value from one cell to the cell directly to the right. If cell values are the same, then a value of “0” is assigned. If there is a positive difference, a value of “1” will be assigned. If there is a negative difference, a value of “-1” is assigned.

If the **DiffVert** filter is applied to remote sensing imagery, surface roughness is emphasized. Linear features with vertical trends are highlighted. When applied to **DEM**s (Digital Elevation Models) or **sliced** elevation data, this has the effect of creating a “contoured” pseudo shaded relief map layer with lighting either from the left or the right of the map layer depending on the colour sequence that you apply.

The optimal colour scheme for map layers generated by the **DiffVert** filter is “-1” = 100% Red, 100% Green, 100% Blue; “0” = 50% Red, 50% Green, 50% Blue; and “1” = 0% Red, 0% Green, 0% Blue. If this colour sequence produces a “reverse” image to your eye (i.e., hollows appear to be hills and hills appear to be hollows) then try reversing this colour sequence.

**DiffDiag**
The **DiffDiag** filter is a uni-directional edge detection filter. Values are assigned based on the difference in value from one cell to the cell below and to the right. If cell values are the same, then a value of “0” is assigned. If there is a positive difference, a value of “1” will be assigned. If there is a negative difference, a value of “-1” is assigned.

If the **DiffDiag** filter is applied to remote sensing imagery, surface roughness is emphasized. Linear features with diagonal trends are highlighted. When applied to **DEM**s (Digital Elevation Models) or **sliced** elevation data, this has the effect of creating a “contoured” pseudo shaded relief map layer with lighting either from the top left or the bottom right of the map layer depending on the colour sequence that you apply.

The optimal colour scheme for map layers generated by the **DiffDiag** filter is “-1” = 100% Red, 100% Green, 100% Blue; “0” = 50% Red, 50% Green, 50% Blue; and “1” = 0% Red, 0% Green, 0% Blue. If this colour sequence
produces a “reverse” image to your eye (i.e., hollows appear to be hills and hills appear to be hollows) then try reversing this colour sequence.

**Details** The **Filter** operation creates a map layer by applying an image processing filter to an input map layer of continuous data such as a remote sensing image or a **DEM** (Digital Elevation Model).

The **Low Pass**, **High Pass 1**, **High Pass 2**, **Laplacian**, and **Sobel** filters use a moving 3x3 window, or kernel, with a set of default weights. You can override the default kernel by creating and specifying a kernel map layer. The size and weighting of the filter window can thus be controlled. The **Kernel** option substitutes the specified kernel map for the default kernel. The kernel map layer can be either of fixed point or floating point type. The **Kernel** modifier cannot be used with the **Sobel** or **Difference** filters.

**How Do I Create a Convolving Kernel for the Filter Operation?**

The **Filter** operation is typically used for three main purposes: To improve interpretability of the data; to aid in automated feature extraction; to remove and/or reduce data “noise”; and to remove anomalies from interpolated data.

The **Low Pass** filter is a value averaging filter used to smooth data. The **High Pass 1**, **High Pass 2**, **Laplacian**, and **Sobel** filters are edge detection and image enhancement filters used to improve image quality, enhance edges, and detect boundaries. The **Difference** filters are directional filters that are used to contour edges, detect linear features, and generate pseudo shaded relief models.

The colour sequence that you apply to a filtered map layer greatly affects the appearance of the results.

**How Do I Apply a Colour Sequence to Enhance Filter Results?**

**What Do I Need?**

Use the **Filter** operation on continuous data sets such as **DEMs** (Digital Elevation Models), satellite imagery, air photographs, sliced elevation models, and interpolated map layers.

**Troubleshooting Error Messages**

Here are some of the most common error messages for the **Filter** operation with suggestions on what to do if you see them:

**Error, the kernel map must be square and have an odd number of rows/columns.**

Rebuild the kernel map or specify another one.
Error, more than one filter has been specified.
Only one filter type can be specified for each Filter operation.

Error, missing filter type.
There is no default filter type, specify a filter type.

Error, the specified filter type does not use a kernel map.
Do not use a kernel map layer with the filter type specified.