```
M_RemovedBackground=(M_RemovedBackground-V_min)/(V_max-V_min)
// Remove zeros by replacing with average value.
WaveStats/Q/M=1 M_RemovedBackground
MatrixOp/O M_RemovedBackground=M_RemovedBackground+V_avg*equal(M_RemovedBackground,0)
MatrixOp/O mulBlobs=mulBlobs/M_RemovedBackground   // scaled image.
```

In the example above we have manually created the ROI masks that were needed for the fit. You can automate this process (and actually improve performance) by subdividing the image into a number of smaller rectangles and selecting in each one the highest (or lowest) pixel values. An example of such procedure is provided in connection with the ImageStats operation above.

General Utilities: ImageTransform Operation

As we have seen above, the **ImageTransform** operation (see page V-417) provides a number of image utilities. As a rule, if you are unable to find an appropriate image operation check the options available under ImageTransform. Here are some examples:

When working with RGB or HSL images it is frequently necessary to access one plane at a time. For example, the green plane of the peppers image can be obtained as follows:

```
NewImage root:images:peppers        // display original
Duplicate/O root:images:peppers peppers
ImageTransform /P=1 getPlane peppers
NewImage M_ImagePlane                // display green plane in grayscale
```

The complementary operation can insert a plane into a 3D wave. For example, suppose you wanted to modify the green plane of the peppers image:

```
ImageHistModification/o M_ImagePlane
ImageTransform /p=1 /D=M_ImagePlane setPlane peppers
NewImage peppers                     // display the processed image
```

Some operations are restricted to waves of particular dimensions. For example, if you want to use the Adaptive histogram equalization, the number of horizontal and vertical partitions is restricted by the requirement that the image be an exact multiple of the dimensions of the subregion. The ImageTransform operation provides three image padding options: If you specify a negative number to the changed rows or columns, the corresponding rows and columns are removed from the image. If the numbers are positive, rows and columns are added. By default the added rows and columns contain exactly the same pixel values as the last row and column in the image. If you specify the /W flag the operation duplicates the relevant portion of the image into the new rows and columns. Here are some examples:

```
Duplicate/o root:images:baboon baboon
NewImage baboon
ImageTransform/N={-20,-10} padImage baboon
Rename M_PaddedImage, cropped
NewImage cropped
ImageTransform/N={40,40} padImage baboon
Rename M_PaddedImage, padLastVals
NewImage padLastVals
ImageTransform/W/N={100,100} padImage baboon
NewImage M_PaddedImage
```

Another utility operation is the conversion of any 2D wave into a normalized (0-255) 8-bit image wave. This is accomplished with the ImageTransform operation using the keyword convert2gray. Here is an example:

```
// Create some numerical data
Make/O/N=(50,80) numericalWave=x*sin(x/10)*y*exp(y/100)
ImageTransform convert2gray numericalWave
NewImage M_Image2Gray
```

The conversion to an 8-bit image is required for certain operation. It is also useful sometimes when you want to reduce the size of your image waves.