

```

// references to the various M_ImagePlane waves, they too are
// automatically deleted.
KillWaves dfw
End

```

To run the demo, execute:

```
Demo(50)
```

On an eight-core Mac Pro, this took 4.1 seconds without the MultiThread keyword and 0.6 seconds with the MultiThread keyword for a speedup of about 6.8 times.

## Wave Reference MultiThread Example

In the preceding example, free data folders were used to hold data processed by threads. Since each free data folder held just a single wave, the example can be simplified by using free waves instead of free data folders. So here we perform the same threaded filtering of planes using free waves.

Because MultiThread is used, multiple instances of Worker execute simultaneously on different cores. Each instance runs in its own thread, working on a different plane. Each instance returns one filtered plane in a free wave named M\_ImagePlane. The use of free waves allows each instance of Worker to work on its own M\_ImagePlane wave without creating a name conflict.

This version of the example relies on the fact that a wave in a free data folder becomes a free wave when the free data folder is automatically deleted. See **Free Wave Lifetime** on page IV-92 for details.

```

ThreadSafe Function/WAVE Worker(w3DIn, plane)
  WAVE w3DIn
  Variable plane

  DFREF dfSav= GetDataFolderDFR()

  // Create a free data folder and set it as the current data folder
  SetDataFolder NewFreeDataFolder()

  // Extract the plane from the input wave into M_ImagePlane.
  // M_ImagePlane is created in the current data folder
  // which is a free data folder.
  ImageTransform/P=(plane) getPlane, w3DIn
  Wave M_ImagePlane          // Created by ImageTransform getPlane

  // Filter the plane
  WAVE wOut= M_ImagePlane
  MatrixFilter/N=21 gauss,wOut

  // Restore the current data folder
  SetDataFolder dfSav

  // Since the only reference to the free data folder created above
  // was the current data folder, there are now no references it.
  // Therefore, Igor has automatically deleted it.
  // Since there IS a reference to the M_ImagePlane wave in the free
  // data folder, M_ImagePlane is not deleted but becomes a free wave.

  return wOut      // Return a reference to the free M_ImagePlane wave
End

Function Demo(numPlanes)
  Variable numPlanes

  // Create a 3D wave and fill it with data
  Make/O/N=(200,200,numPlanes) srcData= (p==(2*r))*(q==(2*r))

```