Subject: New IDL User Questions
Posted by John Piccirillo on Tue, 08 May 2001 17:38:16 GMT
View Forum Message <> Reply to Message

Hello,

I'm new to IDL, but not new to programming. I have
the IDL manuals and Dr. Fanning's excellent book, nevertheless,
I have a few basic questions:

1. Editor Screen
   a. Is there a way to make the Editor full screen or extend
      over some of the other windows? Using resize doesn't do it.
      Does everyone confine themselves to this small window on their
code?
   b. My scrolling mouse will scroll in the Output Log and Variable
Watch
      windows, but not in the Editor Window. Que pasa?

2. Array Operations - Not being used to IDL type of array operations,
   is there a simpler way to do the following?

a.

```
           For I = 0, 199 Do Begin
              For J = 0, 84 Do Begin
                 If (ImageMask[I,J] EQ 1) Then ImageROI[I,J,*] =
ImageS[I,J,*]
                 Else ImageROI[I,J,*] = 0
              EndFor
          EndFor
```

I thought of using the WHERE function as in,
    ROI = Where(ImageMask EQ 1)
but ImageROI[ROI} = ImageS{ROI} leaves out the third dimension.

b. ;blow-up image X 9 For Screen Display

```
 For j = 0,84 Do Begin
      For i = 0,199 Do Begin
  JImage[3*i,3*j]       = ImageS[i,j,[4]]
  JImage[3*i,3*j+1]     = ImageS[i,j,[4]]
  JImage[3*i,3*j+2]     = ImageS[i,j,[4]]
  JImage[3*i+1,3*j]     = ImageS[i,j,[4]]
  JImage[3*i+1,3*j+1] = ImageS[i,j,[4]]
  JImage[3*i+1,3*j+2] = ImageS[i,j,[4]]
  JImage[3*i+2,3*j]     = ImageS[i,j,[4]]
  JImage[3*i+2,3*j+1] = ImageS[i,j,[4]]
  JImage[3*i+2,3*j+2] = ImageS[i,j,[4]]
     EndFor
 EndFor
```

I don't use the EXPAND function because I don't want to interpolate the data.

3. PLOT

I have a couple of plots I want on the same Y Scale, the larger of the two data sets. Presently, I use plot to generate the scale to !y.range, and then test the two ranges and re-plot, as in.

```
 Window, 0, Title = '  P Target;  NPix = ' + string(Fix(NumOnes)), $
   XSize = 350, YSize = 350, XPos = 0, YPos = 0
 Plot, WavL, MeanPT, PSYM = 2, TickLen = 1, XGrid = 1, YGrid = 1
 PTYRange = !y.crange
 Window, 1, Title = '  P BkGnd;  NPix = ' + string(Fix(17000 - NumOnes)), $
       XSize = 350, YSize = 350, XPos = 0, YPos = 375
 Plot, WavL, MeanPB, PSYM = 2, TickLen = 1, XGrid = 1, YGrid = 1
 PBYRange = !y.crange
 SPRange = Max([PTYRange[1], PBYRange[1]])
 MaxY = [0,SPRange]

; replot all with new, uniform Y scale
 Window, 0, Title = 'P Target;  NPix = ' + string(Fix(NumOnes)), $
   XSize = 350, YSize = 375, XPos = 0, YPos = 0
 Plot, WavL, MeanPT, PSYM = 2, TickLen = 1, XGrid = 1, YGrid = 1, YRange =
MaxY
 Window, 1, Title = 'P BkGnd ;  NPix = ' + string(Fix(17000 - NumOnes)), $
       XSize = 350, YSize = 350, XPos = 0, YPos = 375
 Plot, WavL, MeanPB, PSYM = 2, TickLen = 1, XGrid = 1, YGrid = 1, YRange =
MaxY
```

this seems inefficient, what is a better way?

Thanks,

John Piccirillo