
Subject: IDL ROI Objects

Posted by [David B](#) on Tue, 27 Jan 2015 17:28:13 GMT

[View Forum Message](#) <> [Reply to Message](#)

SO I am currently creating ROI objects so I can group sets of points.

Suppose I can have a set of box values, with corner locations,

X1	X2	Y1	y2
18	22	18	22
1	9	3	11
10	20	5	11
15	19	6	12
8	12	12	18
11	15	9	15

Now these boxes are simply bounding boxes from some ellipses.

I then enter the following commands:

```
=====
cgdisplay, 1000, 800
cgplot, box, xrange=[-10, 25], yrange=[0,25], /NODATA
cgcolorfill, [box[0,0], box[1,0], box[1,0], box[0,0] ], [box[2,0], box[2,0], box[3,0], box[3,0]], $
    color = 'steel_blue'

cgcolorfill, [box[0,1], box[1,1], box[1,1], box[0,1] ], [box[2,1], box[2,1], box[3,1], box[3,1]], $
    color = 'orange'

cgcolorfill, [box[0,2], box[1,2], box[1,2], box[0,2] ], [box[2,2], box[2,2], box[3,2], box[3,2]], $
    color = 'red'

cgcolorfill, [box[0,3], box[1,3], box[1,3], box[0,3] ], [box[2,3], box[2,3], box[3,3], box[3,3]], $
    color = 'purple'
cgcolorfill, [box[0,4], box[1,4], box[1,4], box[0,4] ], [box[2,4], box[2,4], box[3,4], box[3,4]], $
    color = 'sea green'
cgcolorfill, [box[0,5], box[1,5], box[1,5], box[0,5] ], [box[2,5], box[2,5], box[3,5], box[3,5]], $
    color = 'red 2'
```

Followed by:

```
TVELLIPSE, 2, 2, 20, 20, 10.0, /DATA
TVELLIPSE, 3, 4, 5, 7, 35.0, /DATA
TVELLIPSE, 5, 2.5, 15, 8, 10.0, /DATA
TVELLIPSE, 3, 1.5, 17, 9, 85.0, /DATA
TVELLIPSE, 3, 1.5, 10, 15, 85.0, /DATA
TVELLIPSE, 3, 1.5, 13, 12, 67.0, /DATA
cgtext, 20, 20, '0', charsize = 2.0, color = 'white'
```

```
cgtext, 5, 7, '1', charsize = 2.0, color = 'white'
cgtext, 15, 8, '2', charsize = 2.0, color = 'white'
cgtext, 17, 9, '3', charsize = 2.0, color = 'white'
cgtext, 10, 15, '4', charsize = 2.0, color = 'white'
cgtext, 13, 12, '5', charsize = 2.0, color = 'white'
```

```
=====
```

This is merely an example, I didn't loop yet as I was conducting tests.

And you get a nice display in multiple colours. Some of these boxes overlap, so I tried to create an 'intersection matrix' if you like, of which boxes collide with others. But this is not very efficient.

I then approached the problem using ROIs. I think I can flattened multiple masks into one and then use region labelling to separate the ellipses into groups, according to their bounding box locations.

I discovered that if I ROId the same groups, I get a different graph of the rectangles using the commands:

```
=====
```

```
image = DBLARR(30, 30)
```

```
;Okay, this doesnt look interesting, but it will do
image = NOISE_SCATTER(image)
```

```
;Open a master mask
m_mask = DBLARR(szim[0], szim[1])
```

```
=====
```

Then using:

```
=====
```

```
roibox = box
roibox[1, *]++
roibox[3, *]++
```

```
=====
```

The above commands are crucial apparently, I must add 1 to my pixel values to recover the ROI mask correctly...

```
=====
```

```
mask = OBJ_NEW('IDLAnROI', [roibox[0, i], roibox[1, i], roibox[1, i], roibox[0, i], roibox[0, i]], $
    [roibox[2, i], roibox[2, i], roibox[3, i], roibox[3, i], roibox[2, i]]) & $
```

```
;mask -> Setproperty, interior = 0 ;This does bugger and all :(
```

```
;Mask must match the image dimensions
```

```
t_mask = mask -> ComputeMask( DIMENSIONS = [szim[0], szim[1]], $
    PIXEL_CENTER = [1, 1], MASK_RULE = 1) & $
```

```
;Add this onto the mask
```

```
m_mask = m_mask + t_mask
```

```
cgdisplay, 900, 900
cgplot, box, xrange = [0, 30], yrange = [0,30], /NODATA, aspect = 1.0
cgimage, m_mask, /keep, /overplot
cgplot, box, xrange = [0, 30], yrange = [0,30], $
  axiscolor='red', /NOERASE , /NODATA, aspect = 1.0
=====
```

If I do not add the single pixel and set the PIXEL_CENTER to subtract a pixel off, then my mask seems to be a pixel down on the Y2 and X2 lines.

My question is:

Does this seem a reasonable solution? Or am I being very stupid? I can understand what the ROI is doing, but a little unsure exactly what the issue is?

Is taking the PIXEL_CENTER keyword out and adding/subtracting one from the box like:

```
x1-- & x2++
y1-- & x3++
```

strictly the same result?

David
