
Subject: Re: Inside rectangle

Posted by [btupper](#) on Tue, 02 Jul 2002 18:58:04 GMT

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

On Tue, 2 Jul 2002 12:53:03 -0400, "Neil Talsania"
<neil.talsania@kodak.com> wrote:

> I am relatively new to IDL , and am wondering how to do the following. I
> want to be able to have user input 4 corner coordiates, then check to see if
> a particular point is inside the rectangle. That is pretty easy, i think.
> But then I want to be able to add additional sets of coordinates to check. I
> want this to be unlimited. Basically allowing the user to add rectangles to
> check until he is ready to stop.

Hello,

This came up a while ago in regards to Delauney triangulation meshes... but what's a triangle but a pointy headed rectangle. The following might help. You should note that you will need David Fanning's LoadData function and Liam Gumley's ImDisp procedure. Also, you might want to rename the object... I wasn't very original when I named it.

Once you have all the code do the following from the command line...

```
IDL> .compile anroigroup__define  
IDL> example
```

I hope this helps.

Ben

```
;-----START  
;-----  
; WhichROI  
;-----  
FUNCTION AnROIGroup::WhichROI, X, Y, Z, $  
count = count, type = type, index = index  
  
;This function returns the IDLanROI (OR the ROIs)  
;that contain the point X,Y,Z Only one point is tested  
;per call. If more than ROI contains the point, then  
;an array of IDLanROI objects is returned  
;  
;X,Y,Z see IDLanROI::ContainsPoints  
; ONLY a single point checked  
;COUNT the number of ROIs found that contain X, Y, Z  
;TYPE an array of the values returned by the
```

```
; IDLanROI::ContainsPoints method
;INDEX the postional index of the IDLanROI
; within the AnROIGroup
```

```
np = n_params()
Case np of
  1: data = x[0:2]
  2: data = [x[0], y[0]]
  3: data = [X[0],Y[0], Z[0]]
EndCase
```

```
For i = 0L, self->Count() -1 DO Begin
```

```
  ROI = Self->Get(Position = i)
  r = ROI->ContainsPoints(data)
```

```
  If r GT 0 Then Begin
    If n_elements(Arr) EQ 0 Then Begin
      Arr = ROI
      type = r
      Index = i
    EndIf Else Begin
      Arr = [Arr,ROI]
      type = [type,r]
      Index = [index,i]
    EndElse
  EndIf
```

```
EndFor
```

```
Count = n_elements(Arr)
Return,Arr
END ;WhichROI
```

```
;-----
; Definition
;-----
PRO AnROIGroup__Define
```

```
struct = {AnROIGroup, $
  Inherits IDLanROIGroup}
```

```
END ;AnROIGroup
```

```
;-----
```

```
; EXAMPLE
```

```
;-----
```

```
PRO Example
```

```
bottom = 32
```

```
loadCT, 0, bottom = bottom
```

```
Tek_Color
```

```
XYZ = LoadData(14)
```

```
Triangulate, XYZ[0,*], XYZ[1,*], tri
```

```
Surf = TriGrid( XYZ[0,*], XYZ[1,*], XYZ[2,*], tri, $  
xgrid = xg, ygrid = yg)
```

```
xRange = [Min(xg), Max(xg)]
```

```
yRange = [Min(yg), Max(yg)]
```

```
ImDisp, Surf, /axis, /erase, xrange = xrange, $  
yrange = yrange, bottom = bottom, $  
Color = 1, Background = 0
```

```
oPlot, XYZ[0,*], XYZ[1,*], psym = 6, color = 2
```

```
Group = OBJ_NEW('AnROIGroup')
```

```
For i = 0, n_elements(Tri)/3 - 1 Do Begin  
index = [tri[*], i], tri[0, i]]  
oPlot, XYZ[0, index], XYZ[1, index], color = 3  
o = OBJ_NEW('IDLAnROI', XYZ[0, index], XYZ[1, index])  
Group->Add, o  
EndFor
```

```
myX = [-110.0]
```

```
myY = [29.0]
```

```
PlotS, myX, myY, psym = 2, color = 4
```

```
Arr = Group->WhichROI(myX, myY, $  
count = count, type = type, index = index)
```

```
If Count GT 0 Then Begin
```

```
Help, arr
```

```
print, count
```

```
print, type
```

```
print, index
```

```
Arr[0] ->GetProperty, Data = Data  
oPlot, Data[0,*], Data[1,*], thick = 2, color = 5  
EndIf
```

```
Obj_Destroy, Group  
END  
;-----END
```
