Subject: Re: TRIGRID
Posted by btupper on Tue, 12 Feb 2002 01:23:15 GMT
View Forum Message <> Reply to Message

On 9 Feb 2002 14:11:51 -0800, baratoux@ipgp.jussieu.fr (Baratoux) wrote:

> I need to modify a little the TRIGRID routine. I`d like to know, when
> I get the interpolated regular grid which points have been used (ie
> which triangle) for the interpolation.
>
> If I need the source code for that , do you know where I can find it
> (not the code for the triangulation - Delaunay, but the code for the
> Trigrid procedure).

I'm not sure if this would solve your problem; at least it doesn't modify TRIGRID. That isn't something I know how to do. It does, however, provide a means of linking an arbitrary location with a particular triangle/polygon.

The object below initializes an IDLanROI object for each triangle. Each of these objects gets stuffed into an IDLanROIgroup object (I suppose any container would do for this purpose.) A new method has been whipped up for the ROI group that returns all of the ROIs that contain the data point specified. More than one ROI will be returned if the point is shared along a boundary or the ROIs overlap in some way.

There is an example to run after compiling the object...

```
IDL> .compile anroigroup__define
% Compiled module: ANROIGROUP::WHICHROI.
% Compiled module: ANROIGROUP__DEFINE.
% Compiled module: EXAMPLE.
IDL> example
```

Hope this helps,

Ben

```
;**********BEGIN HERE

;------
; WhichROI
```

```
;------
FUNCTION AnROIGroup::WhichROI, X, Y, Z, $
 count = count,  type = type, index = index

 ;This function returns the IDLanROI (OR the ROIs)
 ;that contains 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  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 HERE
```