
Subject: Re: Edge detection and convert edges into polygons

Posted by [David Fanning](#) on Fri, 01 Aug 2008 15:06:54 GMT

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

M. Suklitsch writes:

> I need some hint on how to convert edges into vertices. What I want to
> do in the end is check whether a station (of which I have lon/lat
> values) lies within such a vertex.
>
> To be more precise: I have a netCDF file with three fields. longitude,
> latitude and mask, where mask has integer values and defines
> subregions. In this mask "image" I want to detect the edges, what I
> already got working using the tips on http://www.dfanning.com/ip_tips/sharpen.html.
>
> These edges I want to convert into vertices which I then put in an
> IDLanROI object. Using this object I could then very easily check if a
> station given by its longitude and latitude lies within a region or
> not.

I think I would just work with the integer mask and find
the boundaries around the sub-regions:

http://www.dfanning.com/ip_tips/boundary.html

You can use the boundary directly to create an IDLanROI
object, or (probably a good idea) you can reduce the
number of vertices in the boundary with MESH_DECIMATE
first.

Cheers,

David

--

David Fanning, Ph.D.

Fanning Software Consulting, Inc.

Coyote's Guide to IDL Programming: <http://www.dfanning.com/>

Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: Edge detection and convert edges into polygons

Posted by [pgrigis](#) on Fri, 01 Aug 2008 15:11:16 GMT

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

M. Suklitsch wrote:

> Hi!
>
>

> I need some hint on how to convert edges into vertices. What I want to
> do in the end is check whether a station (of which I have lon/lat
> values) lies within such a vertex.
>
> To be more precise: I have a netCDF file with three fields. longitude,
> latitude and mask, where mask has integer values and defines
> subregions. In this mask "image" I want to detect the edges, what I
> already got working using the tips on http://www.dfanning.com/ip_tips/sharpen.html.
>
> These edges I want to convert into vertices which I then put in an
> IDLanROI object. Using this object I could then very easily check if a
> station given by its longitude and latitude lies within a region or
> not.

Still not sure if I understand... do you have a mask wich is built out
of several disconnected pieces and you are trying find out, for a
given
point, in which part of the mask it lies?

Ciao,
Paolo

>
>
> Another possibility would be to simply search the nearest gridpoint to
> my station and say that the region index of this gridpoint is the one
> for my station. But I think the first way is more correct. Does anyone
> have some idea on how to do this? I hope I explained the problem in a
> way that everybody understands it. :)
>
>
> Thanks in advance!
>
> Cheers,
> Martin

Subject: Re: Edge detection and convert edges into polygons
Posted by [M. Suklitsch](#) on Mon, 04 Aug 2008 07:50:25 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi!

Thanks for the quick replies!

Paolo, I have several subregions, defined over a specific area. These

subregions are sometimes bigger (> 100 pixels), sometimes only consist of three or four pixels. And yes, I want to find out in which of these subregions my station, or more general a point of interest, lies.

David, this is what I've come up with thanks to your hints. Although this code works, maybe there is some shortcut that makes the whole thing even better?

```
FOR mm = 1, iNumOfRegs-1 DO BEGIN
    ; do some edge detection in rgiGridMsk
    rgiIndices = WHERE(rgiGridMsk EQ mm)
    IF rgiIndices[0] NE -1 THEN BEGIN
        rgiBoundary = Find_Boundary(rgiIndices,$
            xsize=rgiGridSiz[0],ysize=rgiGridSiz[1])

        rgrXVect = FLTARR(N_ELEMENTS(rgiBoundary)/2) & rgrYVect =
rgrXVect

        ; now get the corresponding lat/lon values and store them as
vectors
        FOR nn = 0, N_ELEMENTS(rgiBoundary)/2 -1 DO BEGIN
            rgrXVect[nn] =
rgrGridLon[rgiBoundary[0,nn],rgiBoundary[1,nn]]
            rgrYVect[nn] =
rgrGridLat[rgiBoundary[0,nn],rgiBoundary[1,nn]]
        ENDFOR

        ; put the polygons into IDLanROI object
        grROI = OBJ_NEW('IDLanROI', rgrXVect, rgrYVect)

        ; use IDLanROI->ContainsPoints to check whether stations lie
; within or without a region
        rgiMatch = grROI->ContainsPoints(rgrLon,rgrLat)
        iFoundMatch = (WHERE(rgiMatch NE 0))[0]
        IF iFoundMatch NE -1 THEN BEGIN
            rgiStationMaskIdx[WHERE(rgiMatch NE 0)] = mm
        ENDIF
    ENDIF
ENDFOR
```

Maybe a short explanation: rgrLon and rgrLat are vectors holding the longitude and latitude of several points of interest, rgiStationMaskIdx is a vector of the same size as rgrLon where the corresponding subregion index is stored.

Cheers,

Martin

David Fanning schrieb:

>
> I think I would just work with the integer mask and find
> the boundaries around the sub-regions:
>
> http://www.dfanning.com/ip_tips/boundary.html
>
> You can use the boundary directly to create an IDLanROI
> object, or (probably a good idea) you can reduce the
> number of vertices in the boundary with MESH_DECIMATE
> first.
>
> Cheers,
>
> David
> --
> David Fanning, Ph.D.
> Fanning Software Consulting, Inc.
> Coyote's Guide to IDL Programming: <http://www.dfanning.com/>
> Sepore ma de ni thui. ("Perhaps thou speakest truth.")
