Subject: Label_region and Erosion
Posted by lbryanNOSPAM on Tue, 03 Nov 1998 08:00:00 GMT
View Forum Message <> Reply to Message

I'm sending the group some sample code (it hasn't been overly debugged or documented, but should work) I've written with some of David Fanning's and Struan (Gray)'s code as a base. I'm looking to create a multi-surface plot, from a 2-D data array. The procedure is straightforward and works great on the simple data set I've set up. I need, however, something that can handle some ambiguous and noisy surfaces. My first question is how do I find out what algorithm is used in LABEL_REGION? Since I do not see this function in the library, I assume it is written in C somewhere. I'm trying to use it to detect surfaces in my target volume and am having mixed results. How does it decide what is a unique surface and what is only a bump on a surface? Thanks for any info you can pass along.

Also, I've had a suggestion to use morphologic filters, erode and dilate. They look helpful for my goal. From the IDL books, I think I see how they work on binary applications, but the greyscale use is confusing me. Does anyone have an example of how they work on this kind of an application?

Lastly, I wan't to apply a median filter (and possible other filters) to some sections of my surfaces but not others (all irregular shapes). I imaging I'll have to write my own procedure where I pass over my data with a filter and a masking function to exclude certain areas. Has anyone already done this? Am I missing an easy way to do this?

Thanks in advance. Here's my code (messy as it is!).

Lisa Bryan

```
PRO MULTI_SURF_EXAMPLE

plane = fltarr(100,100)
plane(55:85,20:50) = dist(31)+10
plane(5:15,40:70) = (findgen(341))/100+30
plane(30:35,5:75) = (findgen(426))/100+20
plane(75:95,75:95) = (findgen(441))/100+25

shade = dist(100)

MULTI_SURF, PLANE, SHADE, MAXSHADE = 70, b

END

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
```

```
PRO MULTI_SURF, bottdepth, bottshade, maxshade = maxshade,$
 b
;+
; NAME:
; MULTI_SURF
; PURPOSE:
; To seperate a single surface into multisurfaces and
; plot them all to a single 3-D SHADE_SURF plot.
;
; CALLING SEQUENCE:
; MULTI_SURF, bottdepth, bottshade
;
; INPUTS:
; bottdepth is a 2d array that is the depth of each pixel
; bottshade is a 2d array with the same dimensions as bottdepth
; which shows the relative intensity of each pixel
;
; INPUT KEYWORDS:
; MAXSHADE: This is the maximum value to be included in the
; shades which are to be plotted over the surface.
;
;
; OUTPUTS:
;
;
; NOTES:
;
; REVISION HISTORY:
; Written E.L. Bryan Nov. 1998
;

;  check the dimensions

if (total(size(bottdepth)) ne total(size(bottshade))) then begin
 print, 'Depth and Shade arrays must be of the same dimension'
 return
endif

image = bottdepth
image = image - min(image) ;set min(image) to 0

szimage = size(image)

b = LABEL_REGION(image,/eight) ;define surfaces in b
h = HISTOGRAM(b, REVERSE_INDICES=r) ;Get population and members of

      ;each blob.
```

```
;set up Z buffer
thisDevice = !D.Name
Set_Plot, 'Z'
Device, Set_Resolution=[szimage(1) > 500,szimage(2) > 400]

;set up axes with no data
surface,image,zrange = [min(image),max(image)],$
  min_value = min(image),/nodata



FOR i=0, N_ELEMENTS(h)-1 DO BEGIN ;Select each region

 current_surf = image
 current_surf(where(b(*) ne i)) = -10 ;place everything but
    ;current surface below plotted region

 shade_surf,current_surf,zrange = [min(image),max(image)],$
  min_value = min(image),/noerase, $
  shades = bytscl(bottshade,max = maxshade)
 ;count regions
 p = r(r[i]:r[i+1]-1) ;Subscripts of members of region i.
 q = image[p] ;Pixels of region i
 PRINT, 'Region ', i, $
  ', Population = ', h[i]
ENDFOR

snapshot = TVRD()
Set_Plot, thisDevice
window,0,xs = szimage(1) > 500,ys = szimage(2) > 400
TV, snapshot
end
```

Arete Associates
Tucson, Arizona
lbryan@arete-az.com