Subject: Help With Finding Local Maxima of an Image (locmax)
Posted by einszweilieb on Wed, 25 Mar 2009 19:35:08 GMT
View Forum Message <> Reply to Message

I'm trying to use R. Sterner's "locmax" function (see below) to find local maxima (peaks of intensities) in an image. I'm a bit of a newbie to IDL so forgive me if my questions are silly :)

As far as I can tell, all I need to do is read in the image (img) and use the command "locmax, img". Is that correct?

When I do this I get the error message which says something like "return statements in functions must have 1 value" Line 79 (i.e. the second to last line below with the "return" statement).

What's going wrong here? Why is the function trying to return multiple values? Isn't that what it's supposed to do - return all of the places where there's a local max?

Can anyone help me with this? Does anyone know of any other procedures/functions which do similar things?

Thanks.

```
 ;------------------------------------------------------------ --
;+
; NAME:
;       LOCMAX
; PURPOSE:
;       Find local maxima in an image.
; CATEGORY:
; CALLING SEQUENCE:
;       locmax, img
; INPUTS:
;       img = image to process.              in
; KEYWORD PARAMETERS:
;       Keywords:
;         MASK=m returns a mask image with 1 at all
;           local maxima and 0 elsewhere.
;         WHERE=w returns 1-d indices of all local maxima.
;           -1 if no local maxima.
;         VALUES=v returns values of img at all local maxima.
;         VALUE_IMAGE=vimg use vimg to determine values.
;           Instead of img.
;         IX=ix returns x index of all local maxima.
;         IY=iy returns y index of all local maxima.
;         /SORT sorts local maxima by descending image values.
```

```
;       /NOEDGE ingores any maxima at image edges.
; OUTPUTS:
; COMMON BLOCKS:
; NOTES:
;       Notes: All output is through keywords.
;         Ignores plateaus.  May not work for
;         all edge points.
; MODIFICATION HISTORY:
;       R. Sterner, 17 Aug, 1990
;       R. Sterner, 27 Aug, 1990 --- added value_image.
;
; Copyright (C) 1990, Johns Hopkins University/Applied Physics
Laboratory
; This software may be used, copied, or redistributed as long as it is
not
; sold and this copyright notice is reproduced on each copy made.
This
; routine is provided as is without any express or implied warranties
; whatsoever.  Other limitations apply as described in the file
disclaimer.txt.
;-
 ;------------------------------------------------------------ --

 function locmax, img, mask=m, where=w, ix=ix, iy=iy, sort=srt, $
   values=v, value_image=vimg, noedge=noed, help=hlp

 fuzz = 1.e-8  ; Ignore values below this.

 ;-----  Shift four ways  -----
 dx1 = shift(img,1,0)
 dx2 = shift(img,-1,0)
 dy1 = shift(img,0,1)
 dy2 = shift(img,0,-1)
 ;------  compare each pixel to 4 surrounding values  -------
 m = (img gt dx1) and (img gt dx2) and (img gt dy1) and (img gt dy2)
 if keyword_set(noed) then imgfrm, m, 0
 ;------  number of local maxima  --------
 w = where(m eq 1, count) ; Find local maxima.
 fzz = img(w)   ; Pull out values.
 wfzz = where(fzz lt fuzz, c) ; Look for values below fuzz.
 if c gt 0 then begin  ; Found any?
   m(w(wfzz)) = 0  ;   Yes, zap them.
   w = where(m eq 1, count) ;   Now try again for local maxima.
 endif
 ;------  if any continue  ------
 if count gt 0 then begin
   if n_elements(vimg) eq 0 then begin ; Pick off values at maxima.
     v = img(w)
```

```
  endif else begin
    v = vimg(w)
  endelse
  if keyword_set(srt) then begin ; Sort?
    is= reverse(sort(v))  ; yes.
    v = v(is)
    w = w(is)
  endif
  one2two, w, img, ix, iy  ; Convert to 2-d indices.
endif

return

end
```