Subject: Maximum ROI within an ROI
Posted by Messon Gbah on Tue, 12 Sep 2000 16:05:22 GMT
View Forum Message <> Reply to Message

Here is a problem which may be trivial for some of you IDL gurus. I will be VERY grateful for any help, help, direction, or pointers.

Take 2 regions (ROIs) of an image: One outer ROI and one inner ROI. The inner ROI fits entirely inside the outer. For example:

```
;inner ROI, call it oROI
ix = [100, 150, 150, 100, 100]
iy = [70, 70, 120, 120, 70]
;outer ROI, call it iROI
ox = [100, 200, 150, 150, 200, 200, 100, 100]
oy = [70, 70, 95, 120, 145, 170, 170, 70]

;and a simple image
image = dist(256)

tv, image               ;display the image
plots, ox, oy, /device ;display oROI
plots, ix, iy, /device ;display iROI
```

Now the problem: Search ONLY the sub-image inside the oROI and find:
1) The number N of possible positions where iROI fits inside the oROI.
2) The optimal ROI (call it maxROI) characterized by the largest total(image(maxROI))/n_elements(iROI) amongst all N possible iROI positions inside oROI. The average value of the sub-image inside maxROI is the is highest of all N possible.

Thanks in advance.

Messon Gbah

Subject: Re: Maximum ROI within an ROI
Posted by Messon Gbah on Fri, 15 Sep 2000 07:00:00 GMT
View Forum Message <> Reply to Message

"Messon Gbah" <gbah@umich.edu> wrote in message news:%vsv5.3053$O5.61755@news.itd.umich.edu...
> Here is a problem which may be trivial for some of you IDL gurus. I will be
> VERY grateful for any help, help, direction, or pointers.
>
> Take 2 regions (ROIs) of an image: One outer ROI and one inner ROI. The
> inner ROI fits entirely inside the outer. For example:

>
> ;inner ROI, call it oROI
> ix = [100, 150, 150, 100, 100]
> iy = [70, 70, 120, 120, 70]
> ;outer ROI, call it iROI
> ox = [100, 200, 150, 150, 200, 200, 100, 100]
> oy = [70, 70, 95, 120, 145, 170, 170, 70]
>
> ;and a simple image
> image = dist(256)
>
> tv, image ;display the image
> plots, ox, oy, /device ;display oROI
> plots, ix, iy, /device ;display iROI
>
> Now the problem: Search ONLY the sub-image inside the oROI and find:
> 1) The number N of possible positions where iROI fits inside the oROI.
> 2) The optimal ROI (call it maxROI) characterized by the largest
> total(image(maxROI))/n_elements(iROI)
> amongst all N possible iROI positions inside oROI. The average value of
the
> sub-image inside maxROI is the is highest of all N possible.
>
> Thanks in advance.
>
> Messon Gbah
>

After some inital guessing, I came up with the following solution. It's not elegant and it VERY slow because of the nested for loops. Hope you can spot obvious improvements in the code.

Messon.

```
 ;------------------------------------------------------------ ----------
---
;Use one of David Fanning's set functions (from David's web site) to get the
;intersection of two sets of roi subscripts (obtained with get_roi)

FUNCTION SetIntersection, a, b

minab = Min(a, Max=maxa) > Min(b, Max=maxb) ;Only need intersection of
ranges
maxab = maxa < maxb

; If either set is empty, or their ranges don't intersect: result = NULL.

IF (maxab LT minab OR maxab LT 0) then return, -1
```

```
r = Where((Histogram(a, Min=minab, Max=maxab) NE 0) AND  $
        (Histogram(b, Min=minab, Max=maxab) NE 0), count)

IF (count EQ 0) then return, -1 else return, r + minab

end

 ;---------------------------------------------------------- ----------------
--
;Use polyfillv to get the subscripts contained in a ROI
;
function get_roi, x, y, npts, zoom, ximgsz, yimgsz

if (npts ge 3) then begin
   xpts = (x(0:npts-1) + zoom-1) / zoom
   ypts = (y(0:npts-1) + zoom-1) / zoom
   subs = polyfillv(xpts, ypts, ximgsz, yimgsz)
endif else begin
   subs = -1
endelse

return, subs
end

 ;---------------------------------------------------------- ----------------
-
pro test_maxroi

zoom = 1
ximgsz = 256 & yimgsz = 256
image = dist(ximgsz)


;define outer ROI oROI
onpts = 7 ;number of points in oROI
ox = [100, 200, 150, 150, 200, 200, 100, 100]
oy = [70, 70, 95, 120, 145, 170, 170, 70]

;define inner ROI iROI
inpts = 5 ;number of points in iROI
ix = [100, 120, 120, 100, 100]
iy = [70, 70, 100, 100, 70]

window,0, xsize = 270, ysize = 270
drawcolor = !d.n_colors-1

tvlct, 255, 255, 0, drawcolor
bimage = bytscl(image, top=drawcolor-1)
```

```
tv, rebin(bimage, ximgsz*zoom, yimgsz*zoom)


plots, ox, oy, /device
plots, ix, iy, /device

;get subscripts of outer ROI, oROI
osubs = get_roi(ox, oy, onpts, zoom, ximgsz, yimgsz) ;get osubs
onpix = n_elements(osubs)

;get subscripts of inner ROI, iROI
isubs = get_roi(ix, iy, inpts, zoom, ximgsz, yimgsz) ;get isubs
inpix = n_elements(isubs)

print, ' inpts = ', inpts, ' onpts = ', onpts
print, 'onpix = ', onpix, ' inpix = ', inpix

oxmin = min(ox, max=oxmax) & oymin = min(oy, max=oymax)
;square bounding box enclosing iROI
ixmin = min(ix, max=ixmax) & iymin = min(iy, max=iymax)


;main loop to scan oROI starts
small = -9999.99
max_roi = small
imax = 0 & jmax = 0 & npos = 0
for j=0, oymax-oymin-1 do begin
  for i=0, oxmax-oxmin-1 do begin
    isubs = get_roi(ix+i, iy+j, inpts, zoom, ximgsz, yimgsz)
    subs = SetIntersection(osubs, isubs)
    if ((n_elements(subs) eq inpix) and $
       ((oxmin le ixmin+i) and (ixmax+i le oxmax) and $
        (oymin le iymin+j) and (iymax+j le oymax))) then begin
      npos = npos + 1
      avge = total(image(isubs))/inpix
    endif else avge = small
    if (avge gt max_roi) then begin
      imax = i
      jmax = j
      max_roi = avge
;       print, 'imax =', imax, '. jmax =', jmax, '. max_roi =', max_roi
      plots, ix+i, iy+j, /device
    endif
  endfor
endfor

;vertex of the max ROI sought
ix = ix + imax
```

```
iy = iy + jmax

print, 'Number of possible positions of iROI withing oROI = ', npos
print, 'vertex of the Max roi within oROI is:'
print, 'ix:', ix
print, 'iy:', iy
ixmin = min(ix, max=ixmax) & iymin = min(iy, max=iymax)
print,'Centered at: imax = ', (ixmin+ixmax)/2, ' jmax =', (iymin+iymax)/2
print, 'Average pix value in max area =', max_roi

;display result
erase
bimage = bytscl(image, top=drawcolor-1)
tv, rebin(bimage, ximgsz*zoom, yimgsz*zoom)

plots, ox, oy, /device
plots, ix, iy, /device

return
end
```
