Subject: ENVI batch ROI
Posted by jdvona on Wed, 05 Jun 2002 15:13:01 GMT
View Forum Message <> Reply to Message

I am not an IDL programmer and need help with a routine I developed. The routine batch processes a GIS vector file with multiple polygons by creating a single ROI for each individual polygon, which then outputs the mean, total and standard deviation for the pixels within the ROI.

The program has worked well for very simple polygons with few verticies, but I found a bug, which occurs with fairly complex (eg real-world) vegetation polygons. After the routine completes if I look at the ROI's in ENVI they have missing parts of the polygons, weird bisections, and filled in holes that were not part of the original polygon. The code which creates the ROI is below:
I think 3.5 has this capability but I am still running 3.2 (long story).
Thanks for the help.

```
FOR i=0,num-1 DO BEGIN

  ; read the record
  ;
  vec = ENVI_EVF_READ_RECORD(evf_id, i)
  ; convert from map coordinates to file coordinates, again
  ; this assumes that the map proj for the vector data is the
  ; same as that of the image
  ;
  ENVI_MAP_CONVERT, xpts, ypts, vec(0,*), vec(1,*), h_map=h_map,
/to_file
  ; make an ROI in memory
  roi_id = ENVI_CREATE_ROI(ns=ns, nl=nl)

  ; add the vector record as an ROI
  ENVI_DEFINE_ROI, roi_id, xpts=REFORM(xpts, /over), $
    ypts=REFORM(ypts, /over), /polygon

  ; extract the image data associated w/the ROI
  ; XXX CHANGE J=0,8 below to represent the appropriate number of
bands
 FOR j=0,0 DO BEGIN
  print, roi_id
  roi_data = ENVI_GET_ROI_DATA( roi_id, fid=fid, pos = [j])
   printf,   lun, strtrim(i,2),  total(roi_data),  MEAN(roi_data)
  ENDFOR
  ENDFOR
```

Subject: Re: ENVI batch ROI
Posted by Peter Scarth on Tue, 11 Jun 2002 04:51:54 GMT
View Forum Message <> Reply to Message

Hi John,
This code seems to work OK in 3.5 with not-to-complex rainforest polygons.
There was a bug in 3.2 that was fixed with SP2:
The user procedure ENVI_DEFINE_ROI incorrectly bounded the incoming XPTS
with number_of_lines-1 instead of number_of_samples-1
The 3.2 service pack 2 seems to have vanished from the rsinc web site, but
you can still get it from http://www.intersys21.com/product/alldown.html
It might help...
Cheers,
Peter

"John" <jdvona@yahoo.com> wrote in message
news:844e4579.0206050713.67b2d0df@posting.google.com...
> I am not an IDL programmer and need help with a routine I developed.
> The routine batch processes a GIS vector file with multiple polygons
> by creating a single ROI for each individual polygon, which then
> outputs the mean, total and standard deviation for the pixels within
> the ROI.
>
> The program has worked well for very simple polygons with few
> verticies, but I found a bug, which occurs with fairly complex (eg
> real-world) vegetation polygons. After the routine completes if I
> look at the ROI's in ENVI they have missing parts of the polygons,
> weird bisections, and filled in holes that were not part of the
> original polygon. The code which creates the ROI is below:
> I think 3.5 has this capability but I am still running 3.2 (long
> story).
> Thanks for the help.

Subject: Re: ENVI batch ROI
Posted by jdvona on Wed, 19 Jun 2002 15:07:13 GMT
View Forum Message <> Reply to Message

"Peter Scarth" <p.scarth@uq.edu.au> wrote in message
news:<ae3vlhiii1@bunyip.cc.uq.edu.au>...
> Hi John,
> This code seems to work OK in 3.5 with not-to-complex rainforest polygons.
> There was a bug in 3.2 that was fixed with SP2:
> The user procedure ENVI_DEFINE_ROI incorrectly bounded the incoming XPTS
> with number_of_lines-1 instead of number_of_samples-1

> The 3.2 service pack 2 seems to have vanished from the rsinc web site, but
> you can still get it from http://www.intersys21.com/product/alldown.html
> It might help...
> Cheers,
> Peter

Thanks for the help, but the patch didn't improve the situation. I
learned that the bug only occurs with multi-part polygons, e.g. veg
polygons that are bisected by roads, utilities right-of-ways etc,
which creates multiple pieces.
However, if I take an individual multi-part polygon, open it up in
ENVI, and convert it to an ROI using the ENVI widget, it works fine.
This means that something is obviously wrong with ENVI_DEFINE_ROI or
ENVI_MAP_CONVERT.

Too bad RSI doesn't let us peak at the code behind their functions so
I can fix this error. Is this corrected in 3.5?
John

Subject: Re: ENVI batch ROI
Posted by Peter Scarth on Tue, 25 Jun 2002 05:36:23 GMT
View Forum Message <> Reply to Message

Hi John,
I can now see what the problem is - ENVI_DEFINE_ROI does not test for
multipart polygons. You have to work this out for yourself by adding in the
following lines. There may be a simpler way, but this method seems to work
with complex vegetation multipart polygons.

This raises an interesting point - how can you detect multipart lines in a
EVF?

Cheers,
Peter

> FOR i=0,num-1 DO BEGIN
>
> ; read the record
> ;
> vec = ENVI_EVF_READ_RECORD(evf_id, i)
> ; convert from map coordinates to file coordinates, again
> ; this assumes that the map proj for the vector data is the
> ; same as that of the image
> ;

```
>    ENVI_MAP_CONVERT, xpts, ypts, vec(0,*), vec(1,*), h_map=h_map,
>  /to_file
>     ; make an ROI in memory
>     roi_id = ENVI_CREATE_ROI(ns=ns, nl=nl)
>

; -------------Start new ENVI_DEFINE_ROI section --------------

num_vert=n_elements(xpts)-1 ; How many vertices in the record?
sv=0 ; Set the starting vertex index to 0

WHILE (sv lt num_vert-1) DO BEGIN ; Loop until all polygons processed

 ; Find the place where the polygon closes
 idx=where((xpts[sv] eq xpts[sv+1:num_vert])$
    AND (ypts[sv] eq ypts[sv+1:num_vert]),match)


 IF match eq 0 THEN BREAK ; Exit if it is not a polygon

 ; If there is closure start processing
 ed=sv+1+idx[0] ; Find the array index of the endpoint vertex

 ; Write the polygon
 ENVI_DEFINE_ROI, roi_id, xpts=REFORM(xpts[sv:ed], /over),$
     ypts=REFORM(ypts[sv:ed], /over), /polygon

 sv=ed+1 ; Set the new start vertex index

ENDWHILE

; -------------End new ENVI_DEFINE_ROI section --------------

>
>     ; extract the image data associated w/the ROI
>     ; XXX CHANGE J=0,8 below to represent the appropriate number of
>  bands
>  FOR j=0,0 DO BEGIN
>  print, roi_id
>  roi_data = ENVI_GET_ROI_DATA( roi_id, fid=fid, pos = [j])
>     printf,   lun, strtrim(i,2),  total(roi_data),  MEAN(roi_data)
>     ENDFOR
>     ENDFOR
```

"John" <jdvona@yahoo.com> wrote in message

news:844e4579.0206050713.67b2d0df@posting.google.com...
> I am not an IDL programmer and need help with a routine I developed.
> The routine batch processes a GIS vector file with multiple polygons
> by creating a single ROI for each individual polygon, which then
> outputs the mean, total and standard deviation for the pixels within
> the ROI.
>
> The program has worked well for very simple polygons with few
> verticies, but I found a bug, which occurs with fairly complex (eg
> real-world) vegetation polygons. After the routine completes if I
> look at the ROI's in ENVI they have missing parts of the polygons,
> weird bisections, and filled in holes that were not part of the
> original polygon. The code which creates the ROI is below:
> I think 3.5 has this capability but I am still running 3.2 (long
> story).
> Thanks for the help.
>
> FOR i=0,num-1 DO BEGIN
>
> ; read the record
> ;
> vec = ENVI_EVF_READ_RECORD(evf_id, i)
> ; convert from map coordinates to file coordinates, again
> ; this assumes that the map proj for the vector data is the
> ; same as that of the image
> ;
> ENVI_MAP_CONVERT, xpts, ypts, vec(0,*), vec(1,*), h_map=h_map,
> /to_file
> ; make an ROI in memory
> roi_id = ENVI_CREATE_ROI(ns=ns, nl=nl)
>
> ; add the vector record as an ROI
> ENVI_DEFINE_ROI, roi_id, xpts=REFORM(xpts, /over), $
> ypts=REFORM(ypts, /over), /polygon
>
> ; extract the image data associated w/the ROI
> ; XXX CHANGE J=0,8 below to represent the appropriate number of
> bands
> FOR j=0,0 DO BEGIN
> print, roi_id
> roi_data = ENVI_GET_ROI_DATA(roi_id, fid=fid, pos = [j])
> printf, lun, strtrim(i,2), total(roi_data), MEAN(roi_data)
> ENDFOR
> ENDFOR

Subject: Re: ENVI batch ROI

Posted by jdvona on Tue, 25 Jun 2002 12:54:26 GMT
View Forum Message <> Reply to Message

Peter,
Brilliant! Thanks so much for the help. I hope I never have to detect
multipart lines in an EVF file, I'll just buffer the lines and call it
a polygon ;}
John

"Peter Scarth" <p.scarth@gmx.net> wrote in message
news:<af8vjv$2f5$1@bunyip.cc.uq.edu.au>...
> Hi John,
> I can now see what the problem is - ENVI_DEFINE_ROI does not test for
> multipart polygons. You have to work this out for yourself by adding in the
> following lines. There may be a simpler way, but this method seems to work
> with complex vegetation multipart polygons.
>
> This raises an interesting point - how can you detect multipart lines in a
> EVF?
>
> Cheers,
> Peter