Subject: Re: multiple entities in shapefile
Posted by Klemen on Fri, 26 Nov 2010 11:52:28 GMT
View Forum Message <> Reply to Message

This is an example how to write polygons (pixel positions for MODIS images). Perhaps it hels.

```
PRO modis_geolocation, in_filegeo

 ; Restore global data
 global_data
 restore, 'data.sav'
 ; Open EOS-HDF
 i_fidgeo = EOS_SW_OPEN(in_filegeo, /READ)
 if i_fidgeo eq -1 then begin
    print, 'The input gelocation file does not exist or is not EOS
HDF format!'
    stop
 endif
 i_NSwathgeo = EOS_SW_INQSWATH(in_filegeo, s_SwathListgeo)
 i_swathIDgeo = EOS_SW_ATTACH(i_fidgeo, s_SwathListgeo)     ;attach
object
 i_status_read = EOS_SW_READFIELD(i_swathIDgeo, "Longitude", m_lon)
 ;read longitude
 i_status_read = EOS_SW_READFIELD(i_swathIDgeo, "Latitude", m_lat)
  ;read latitude
 i_status_read = EOS_SW_READFIELD(i_swathIDgeo, "SensorZenith",
m_zen)      ;read longitude
  i_status_read = EOS_SW_READFIELD(i_swathIDgeo, "SensorAzimuth",
m_azm)     ;read latitude
 i_status_detach = EOS_SW_DETACH(i_swathIDgeo)
 i_status_close = EOS_SW_CLOSE(i_fidgeo)
 i_array_size = size(m_lat)
 i_col = i_array_size[1]
 i_lin = i_array_size[2]

;WWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWW
WWWWWWWWWWWWWWW
WWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWW
WWWWWWWWWWWWWWWW WWWWWWWWWWW


;WWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWW
WWWWWWWWWWWWWWW
WWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWW
WWWWWWWWWWWWWWWW WWWWWWWWWWW
 ;Define attributes of ESRI shape file
 n_shp = in_filegeo + 'swath.shp'             ;ime
```

```
  mynewshape=OBJ_NEW('IDLffShape', n_shp, /UPDATE,
ENTITY_TYPE=5)       ;nov polygon shape
  mynewshape->IDLffShape::AddAttribute, 'longitude', 5, 6,
PRECISION=2  ;0
  mynewshape->IDLffShape::AddAttribute, 'latitude', 5, 6,
PRECISION=2   ;1
  mynewshape->IDLffShape::AddAttribute, 'sat_zenith', 5, 6,
PRECISION=2 ;2
  mynewshape->IDLffShape::AddAttribute, 'sat_azimuth', 5, 6,
PRECISION=2;3
  entNew = {IDL_SHAPE_ENTITY}
  attrNew = mynewshape->IDLffShape::GetAttributes(/
ATTRIBUTE_STRUCTURE)
 ;Write shape file
 indx = 0L
 j1 = 500L
 j2 = 600L
 i1 = 1L
 i2 = 1352L
 m_coordinates = make_Array(i_col, i_lin, 8)
 dy = make_Array(i_col)
 for j=j1,j2 do begin
   print, j
   y1 = (j / 10L) * 10L
   y2 = y1 + 9L
   dy[i1:i2] = (m_lat[i1:i2,y1] - m_lat[i1:i2,y2]) / 10.
   ;for i=2L,i_col-3L do begin
   for i=i1,i2 do begin
    ;compute corner coordinates for each pixel
    m_coordinates[i,j,3] = (m_lon[i,j] + m_lon[i-1,j] + m_lon[i,j-1]
+ m_lon[i-1,j-1]) *0.25
    m_coordinates[i,j,2] = (m_lon[i,j] + m_lon[i+1,j] + m_lon[i,j-1]
+ m_lon[i+1,j-1]) *0.25
    m_coordinates[i,j,1] = (m_lon[i,j] + m_lon[i+1,j] + m_lon[i,j+1]
+ m_lon[i+1,j+1]) *0.25
    m_coordinates[i,j,0] = (m_lon[i,j] + m_lon[i-1,j] + m_lon[i,j+1]
+ m_lon[i-1,j+1]) *0.25
    m_coordinates[i,j,7] = (m_lat[i,j] + m_lat[i-1,j] + dy[i]) *0.5
    m_coordinates[i,j,6] = (m_lat[i,j] + m_lat[i+1,j] + dy[i]) *0.5
    m_coordinates[i,j,5] = (m_lat[i,j] + m_lat[i+1,j] - dy[i]) *0.5
    m_coordinates[i,j,4] = (m_lat[i,j] + m_lat[i-1,j] - dy[i]) *0.5

    ;SHAPE
    JUMP1: entNew.ISHAPE = indx
    entNew.SHAPE_TYPE = 5            ;geometry
    entNew.BOUNDS[0] = min(m_coordinates[i,j,0:3])
    entNew.BOUNDS[1] = min(m_coordinates[i,j,4:7])
    entNew.BOUNDS[2] = 0.00000000
```

```
    entNew.BOUNDS[3] = 0.00000000
    entNew.BOUNDS[4] = max(m_coordinates[i,j,0:3])
    entNew.BOUNDS[5] = max(m_coordinates[i,j,4:7])
    entNew.BOUNDS[6] = 0.00000000
    entNew.BOUNDS[7] = 0.00000000
    entNew.N_VERTICES = 5
    v_vertices = [reform(m_coordinates[i,j,0:3], 1,4), $
            reform(m_coordinates[i,j,4:7], 1,4)]
    v_vertices = [[v_vertices], [m_coordinates[i,j,
0],m_coordinates[i,j,4]]]
    p_pointer = ptr_new(v_vertices)
    entNew.VERTICES = p_pointer
    attrNew.ATTRIBUTE_0 = m_lon[i,j]          ;atributes
    attrNew.ATTRIBUTE_1 = m_lat[i,j]
    attrNew.ATTRIBUTE_2 = m_zen[i,j]
    attrNew.ATTRIBUTE_3 = m_azm[i,j]
    mynewshape->IDLffShape::PutEntity, entNew
    mynewshape->IDLffShape::SetAttributes, indx, attrNew
    indx++
  endfor
 endfor

 ;Close shapefile
 OBJ_DESTROY, mynewshape

end
```

Subject: Re: multiple entities in shapefile
Posted by PimK on Fri, 26 Nov 2010 13:32:04 GMT
View Forum Message <> Reply to Message

On Nov 26, 7:52 am, Klemen <klemen.zak...@gmail.com> wrote:
> This is an example how to write polygons (pixel positions for MODIS
> images). Perhaps it hels.
>
> PRO modis_geolocation, in_filegeo
>
>   ; Restore global data
>   global_data
>   restore, 'data.sav'
>   ; Open EOS-HDF
>   i_fidgeo = EOS_SW_OPEN(in_filegeo, /READ)
>   if i_fidgeo eq -1 then begin
>   print, 'The input gelocation file does not exist or is not EOS
> HDF format!'
>   stop
>   endif

```
>       i_NSwathgeo = EOS_SW_INQSWATH(in_filegeo, s_SwathListgeo)
>       i_swathIDgeo = EOS_SW_ATTACH(i_fidgeo, s_SwathListgeo)
                ;attach
> object
>       i_status_read = EOS_SW_READFIELD(i_swathIDgeo, "Longitude", m_lon)
>         ;read longitude
>       i_status_read = EOS_SW_READFIELD(i_swathIDgeo, "Latitude", m_lat)
>             ;read latitude
>       i_status_read = EOS_SW_READFIELD(i_swathIDgeo, "SensorZenith",
> m_zen)       ;read longitude
>   i_status_read = EOS_SW_READFIELD(i_swathIDgeo, "SensorAzimuth",
> m_azm)     ;read latitude
>       i_status_detach = EOS_SW_DETACH(i_swathIDgeo)
>       i_status_close = EOS_SW_CLOSE(i_fidgeo)
>   i_array_size = size(m_lat)
>   i_col = i_array_size[1]
>   i_lin = i_array_size[2]
>
;WWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWW
WWWWWWWWWWWWWWW
WWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWW
WWWWWWWWWWWWWWWW WWWWWWWWWWWW
>
>
;WWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWW
WWWWWWWWWWWWWW
WWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWWW
WWWWWWWWWWWWWWWW WWWWWWWWWWWW
>       ;Define attributes of ESRI shape file
>       n_shp = in_filegeo + 'swath.shp'            ;ime
>   mynewshape=OBJ_NEW('IDLffShape', n_shp, /UPDATE,
> ENTITY_TYPE=5)      ;nov polygon shape
>   mynewshape->IDLffShape::AddAttribute, 'longitude', 5, 6,
> PRECISION=2  ;0
>   mynewshape->IDLffShape::AddAttribute, 'latitude', 5, 6,
> PRECISION=2   ;1
>   mynewshape->IDLffShape::AddAttribute, 'sat_zenith', 5, 6,
> PRECISION=2 ;2
>   mynewshape->IDLffShape::AddAttribute, 'sat_azimuth', 5, 6,
> PRECISION=2;3
>   entNew = {IDL_SHAPE_ENTITY}
>   attrNew = mynewshape->IDLffShape::GetAttributes(/
> ATTRIBUTE_STRUCTURE)
>       ;Write shape file
>       indx = 0L
>       j1 = 500L
>       j2 = 600L
>       i1 = 1L
```

```
>        i2 = 1352L
>        m_coordinates = make_Array(i_col, i_lin, 8)
>        dy = make_Array(i_col)
>        for j=j1,j2 do begin
>     print, j
>     y1 = (j / 10L) * 10L
>     y2 = y1 + 9L
>     dy[i1:i2] = (m_lat[i1:i2,y1] - m_lat[i1:i2,y2]) / 10.
>     ;for i=2L,i_col-3L do begin
>     for i=i1,i2 do begin
>       ;compute corner coordinates for each pixel
>       m_coordinates[i,j,3] = (m_lon[i,j] + m_lon[i-1,j] + m_lon[i,j-1]
> + m_lon[i-1,j-1]) *0.25
>       m_coordinates[i,j,2] = (m_lon[i,j] + m_lon[i+1,j] + m_lon[i,j-1]
> + m_lon[i+1,j-1]) *0.25
>       m_coordinates[i,j,1] = (m_lon[i,j] + m_lon[i+1,j] + m_lon[i,j+1]
> + m_lon[i+1,j+1]) *0.25
>       m_coordinates[i,j,0] = (m_lon[i,j] + m_lon[i-1,j] + m_lon[i,j+1]
> + m_lon[i-1,j+1]) *0.25
>       m_coordinates[i,j,7] = (m_lat[i,j] + m_lat[i-1,j] + dy[i]) *0.5
>       m_coordinates[i,j,6] = (m_lat[i,j] + m_lat[i+1,j] + dy[i]) *0.5
>       m_coordinates[i,j,5] = (m_lat[i,j] + m_lat[i+1,j] - dy[i]) *0.5
>       m_coordinates[i,j,4] = (m_lat[i,j] + m_lat[i-1,j] - dy[i]) *0.5
>
>       ;SHAPE
>       JUMP1: entNew.ISHAPE = indx
>       entNew.SHAPE_TYPE = 5            ;geometry
>       entNew.BOUNDS[0] = min(m_coordinates[i,j,0:3])
>       entNew.BOUNDS[1] = min(m_coordinates[i,j,4:7])
>       entNew.BOUNDS[2] = 0.00000000
>       entNew.BOUNDS[3] = 0.00000000
>       entNew.BOUNDS[4] = max(m_coordinates[i,j,0:3])
>       entNew.BOUNDS[5] = max(m_coordinates[i,j,4:7])
>       entNew.BOUNDS[6] = 0.00000000
>       entNew.BOUNDS[7] = 0.00000000
>       entNew.N_VERTICES = 5
>       v_vertices = [reform(m_coordinates[i,j,0:3], 1,4), $
>               reform(m_coordinates[i,j,4:7], 1,4)]
>       v_vertices = [[v_vertices], [m_coordinates[i,j,
> 0],m_coordinates[i,j,4]]]
>       p_pointer = ptr_new(v_vertices)
>       entNew.VERTICES = p_pointer
>       attrNew.ATTRIBUTE_0 = m_lon[i,j]          ;atributes
>       attrNew.ATTRIBUTE_1 = m_lat[i,j]
>       attrNew.ATTRIBUTE_2 = m_zen[i,j]
>       attrNew.ATTRIBUTE_3 = m_azm[i,j]
>       mynewshape->IDLffShape::PutEntity, entNew
>       mynewshape->IDLffShape::SetAttributes, indx, attrNew
```

```
>     indx++
>    endfor
>  endfor
>
>  ;Close shapefile
>  OBJ_DESTROY, mynewshape
>
> end
```

Excellent! Thanks Klemen your example showed my what I was doing
wrong. I did'nt update the iSHAPE field, so all entries were
overwritten.
Thanks again for sharing your code.
Cheers.