
Subject: Re: problem with subset one image by another

Posted by [Jean H.](#) on Fri, 28 Mar 2008 16:22:13 GMT

[View Forum Message](#) <> [Reply to Message](#)

negra wrote:

wrote:

>> negra wrote:

>>>> Without more details about what you want to do it is difficult,
>>>> bordering upon impossible, for us to give you any useful advice about
>>>> how to do it.

>>> here is idl routine what I written. The beginning of it, is working.

>>> pro spat_subset

>>> cd, 'C:\Scandata\L1a'

>>> HKMfiles = FILE_SEARCH('MOD02HKM.*.img',count=numfiles)

>>> PRINT, '# NDSI files:',N_ELEMENTS(FILE_SEARCH('MOD02HKM.*.img'))

>>> print, FILE_SEARCH(HKMfiles)

>>> for j=0,numfiles-1 DO BEGIN

>>> HKM_name = HKMfiles[j]

>>> print,HKM_name

>>> ;first restore all base save files

>>> ;

>>> envi, /restore_base_save_files

>>> ;

>>> ;Initialize ENVI and send all errors and warnings to the file

>>> batch.txt

>>> ;

>>> envi_batch_init, log_file='batch.txt'

>>> ;

>>> ;Open the input files

>>> ;kz_hkm mask file

>>> envi_open_file, 'C:\Scandata\L1A\mask\kz_hkm', r_fid=file1_fid

>>> if (file1_fid eq -1) then begin

>>> envi_batch_exit

>>> return

>>> endif

>>> envi_open_file,'C:\Scandata\L1A\' + HKMfiles[j], r_fid=file1_fid

>>> if (file1_fid eq -1) then begin

>>> envi_batch_exit

>>> return

>>> endif

>>> envi_file_query, file1_fid, dims=file1_dims, ns=file1_ns, nl=file1_nl,

>>> nb=file1_nb

>>> file1_dims = [-1L,0,file1_ns-1,0,file1_nl-1]

>>> file1_mapinfo = envi_get_map_info(fid=file1_fid)

>>> print, file1_mapinfo

>>> file1_xf = [0,file1_ns-1]

>>> file1_yf = [0,file1_nl-1]

```

>>> envi_convert_file_coordinates, file1_fid, file1_xf, file1_yf,
>>> file1_xmap, file1_ymap, /to_map
>>> print, 'UL corner:',file1_xmap[0],file1_ymap[0]
>>> print, 'LR corner:',file1_xmap[1],file1_ymap[1]
>>> ;Longitude 44.39011111 - 88.38219444
>>> ;Latitude 36.20264722 - 56.35832500
>>> ;subset #1
>>> envi_file_query, file_fid, dims=file_dims, ns=file_ns, nl=file_nl,
>>> nb=file_nb
>>> file_dims = [-1L,0,file_ns-1,0,file_nl-1]
>>> file_mapinfo = envi_get_map_info(fid=file_fid)
>>> print, file_mapinfo
>>> pos = lindgen(file_nb)
>>> out_namea = HKMfiles[j]+'subset.img'
>>> file_mapinfo = envi_get_map_info(fid=file_fid)
>>> _xfactor = file1_mapinfo.ps[0]/file_mapinfo.ps[0]
>>> _yfactor = file1_mapinfo.ps[1]/file_mapinfo.ps[1]
>>> print, [_xfactor, _yfactor]
>>> file_xf = [0,file_ns-1]
>>> file_yf = [0,file_nl-1]
>>> envi_convert_file_coordinates, file_fid, file_xf, file_yf, file_xmap,
>>> file_ymap, /to_map
>> you have to do the opposite: take the projected coordinates of the mask
>> image, and convert it to the Cartesian coordinate of the image to subset.
>>
>> then, don't do the "resize", as you are just changing the pixel size and
>> not the covered area. do subset =
>> image[maskXmin:maskXmax,maskYmin:maskYmax]
>>
>> Jean
>>
>>
>>
>>> print, 'UL corner:',file_xmap[0],file_ymap[0]
>>> print, 'LR corner:',file_xmap[1],file_ymap[1]
>>> ;Longitude 63.29925556 - 106.42788056
>>> ;Latitude 42.01251944 - 64.14574167
>>> ; I have coordinates of image corners
>
> It's write that something wrong but I don't know how will be right.
> Where is mistake?
>
> pos = lindgen(file_nb)
> input = ENVI_GET_DATA(fid=file_fid, dims=file_dims, POS=pos)
> ;
> subset = input[file1_xf[0]:file1_xf[1], file1_yf[0]:file1_yf[1]]

```

This will not work, as file1_xf contains the Cartesian coordinate of the

mask.
first, do

```
envi_convert_file_coordinates, subset_fid, subset_xf, subset_yf,  
Mask_xmap, Mask_ymap
```

so with the above, you have, with respect to the subset image, the
Cartesian coordinates of the Mask corners.

Then do

```
subset = input[subset_xf[0]:subset_xf[1],subset_yf[0]:subset_yf[1]]
```

```
> out_names = 'subset.img'  
> ENVI_WRITE_ENVI_FILE, subset, DATA_TYPE=4 , OUT_NAME=out_names,  
> R_FID=s_fid
```

ENVI_WRITE_ENVI_FILE requires the other following keywords:

NB=integer | NL=integer | NS=integer

OFFSET=value

Jean
