
Subject: Re: How to apply a vegetation index on multiple images?

Posted by [Phillip Bitzer](#) on Mon, 03 Nov 2014 15:46:38 GMT

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

On Sunday, November 2, 2014 12:06:02 PM UTC-6, Muhammad Hasan wrote:

> Hi All,

> I have a vegetation index (Tasseled cap transformation). I have 10 folders for different areas and each folder has 20 Landsat images. My friend helped me to write following code:

I think you need to have a talk with your friend. There are a few things that should be cleaned up. I'll highlight a few lines. One of the nice things about IDL is that it is relatively flexible - you can do things the wrong way and still the right answer. But, it's good to learn the right way now!

> if N_elements(infile) eq -1 then return

I don't think this will ever be true. If this is supposed to catch when the user cancels out the dialog, you should check for the null string.

> cd, path

Honestly, I hate programs that change my working directory (and don't put it back), but it doesn't really hurt you. (But watch out if you start to save things like images/plots/etc - the files will likely end up somewhere you're not looking.)

> envi_open_file, infile(i), r_fid=fid

The first of many like this. Using () to index arrays is very old school and is considered wrong now. Use brackets [] to index arrays. See http://exelisvis.com/docs/Understanding_Array_Subs.html

> file = DIALOG_PICKFILE(FILTER='*.txt')

So, you have to pick a file for every infile? Which means you have to do this 200 times? Crikey, that would drive me crazy.

> WHILE NOT EOF(lun) DO BEGIN & \$

Why is the &\$ there?

> READF, lun, para

> ENDWHILE

So, your while loop goes through the entire file. At every iteration, it reads in a 6x6 float array...until it reaches the end. The variable para will only be the *last* 6x6 array in the file when you move the next step, Is that what you want?

> tct(*,*,0)=float((para(0,0)*b1)+(para(1,0)*b2)+(para(2,0)*b3)+(para(3,0)*b4)+(para(4,0)*b5)+(para(5,0)*b6))

I'm quite sure there are more efficient ways to do this. This looks like a matrix multiplication problem.

```
> filename = FILE_BASENAME(infile(i))
> outfiledir=file_dirname(infile(i),/MARK_DIRECTORY)
> pointPos = STRPOS(filename,'./REVERSE_SEARCH)
> IF pointPos[0] NE -1 THEN BEGIN
```

Hey! Square bracket! Alright! ;-)

> What should I do to make this code capable of computing TCT index automatically? (I am not good at programming)

I haven't the foggiest idea what a TCT index is. You'll have to tell us what it is before we can calculate it.

There are a lot of good resources out there to help you get better at programming. This newsgroup, IDL Data point blog, the online help from Excelsis, Coyote's website (idlcoyote.com), and others I'm not thinking of right now, I'm sure.
