Subject: Re: Convert planetary sinusoidal map image to (lat,lon,value) array
Posted by David Fanning on Mon, 29 Jul 2002 18:27:08 GMT
View Forum Message <> Reply to Message

Ken Mankoff (mankoff@I.HATE.SPAM.cs.colorado.edu) writes:

> I guess I'll give this a shot. Never done this but here is what I
> would try first if I were doing it:

Not too bad, not too bad. :-)

It would certainly work something like this, *if* you could
get an IDL map projection to work. I have my doubts about
what a "sinusoidal equal area" projection is, but assuming it
is the same thing as a sinusoidal projection in IDL, I would
make a few modifications to your suggestions.

> 1) Set up an identical map projection. The key word (no pun intended)
> here is *identical*, so that every pixel of your new "sinusoidal equal
> area" projection is the same as the BYTARR you have been given. If
> you can't get it identical, I am not sure what to do next.
>
> You can test if it is identical via:
> IDL> WINDOW, XSIZE=1440, YSIZE=720
> IDL> MAP_SET, 0, 0, /SIN, /ISO
> IDL> TV, sinusoidal_projection
> IDL> MAP_GRID & MAP_CONTINENTS & MAP_HORIZON
>
> Does the border created by MAP_HORIZON line up exactly (to the pixel)
> with the data your data from the TV command? Look up the keywords to
> MAP_SET if not, and try other projections...

I would add MARGIN=0 and NOBORDER=1 keywords to the MAP_SET
command. And if I knew (or could figure out) the lat/lon coordinates
of the corners of the image I would add the 8-element version of the
LIMIT keyword as well. Then there is a reasonably good chance the
map might match the image.

You could set up the map coordinates in a pixmap if you didn't want
to see something happening on the display. Just make the pixmap the
same size as your image, etc.

> 2) Redo the above code, but stop after the TV command so its just your
> data, nothing extra. You don't actually need a window, but you need
> the MAP_SET command to be run (with the correct 1440,720 sizes) so
> that IDL defines the map coordinate system. You are going to use this
> later to convert between (x,y) pixels and (lat,lon) degrees.
>

> 3) Set up the new array you want to put your data into. I suggest a
> cylindrical 'projection', which can then be warped to any other
> projection you want. So...
> IDL> new_array = BYTARR(1440, 720)

I would just leave it in its current projection, assuming
this is the correct one.

> 4) Step through every (x,y) pixel in your image.

Since the original poster indicated that he was only
interested in non-zero values, I would just work with
those.

```
> FOR x=0,1439 DO BEGIN
>     FOR y=0,719 DO BEGIN
>  aPixel = sinusoidal( x, y )
>  IF ( aPixel NE 0 ) THEN BEGIN ; not a valid (lat,lon) coord.
>      latlon = CONVERT_COORD( x, y, /device, /to_data )
>      new_array[ latlon[0], latlon[1] ] = aPixel
>         ENDIF
>     ENDFOR
> ENDFOR
```

I would do it something like this:

```
 indices = Where(sinusoidal_projection GT 0, count)
 IF count EQ 0 THEN Message, 'Whoops. Somethin gone wrong!!'
 x = indices MOD 1439
 y = indices / 1439
 latlon = CONVERT_COORD( x, y, /DEVICE, /TO_DATA )
```

Now, the information you want is in latlon.

Cheers,

David

--
David W. Fanning, Ph.D.
Fanning Software Consulting, Inc.
Phone: 970-221-0438, E-mail: david@dfanning.com
Coyote's Guide to IDL Programming: http://www.dfanning.com/
Toll-Free IDL Book Orders: 1-888-461-0155