Subject: Re: satellite data
Posted by David Fanning on Tue, 13 May 2003 23:31:03 GMT
View Forum Message <> Reply to Message

Matt McCabe (mmccabe@princeton.edu) writes:

> I know this is an old question, but I'm a new poster so please be
> gentle.

Well, even the old questions are hard questions sometimes.

> I muck around with all sorts of satellite data and am interested in
> finding a nice simple method of plotting said data onto a map
> projection or regular grid wihtout using ENVI. Some of the data I use
> (TRMM-TMI) is irregular in that I just have lat/lon and the associated
> data values. Others, such as when I extract info from TERRA or AQUA
> files, is regular data, but only has the bounding co-ords of the
> image.

TRMM data is more difficult, as I am sure you
know. Liam Gumley has written a nice ImageMap
application that can be used quite nicely for
viewing data like this, although he advices you to
be careful *relying* on the data after you have
produced the map.

   http://cimss.ssec.wisc.edu/~gumley/imagemap.html

For an application I wrote for a client who wanted
to "zoom" into the TRMM data on a map, I had to treat
each piece of data as a polygon that I could stretch
on the map as needed. That worked well enough for our
purposes, but again I probably wouldn't trust interrogating
the actual image values.

If you know the bounding coordinates of the image
you are in pretty good shape. Typically you will
use the 8-element version of the LIMIT keyword to
set up your map coordinate space. This will describe
the lat/lon coordinates of the four corners of the image
(although the documentation is screwy, so you may have to
play around a bit until you get it straight).

It is easiest, probably, to fill an entire window with
the image, then put the map coordinates all the way to the
edge of the window by setting the POSITION keyword to its limits.
You can have resizeable images, etc. with this kind of set-up,
if you use something like TVIMAGE or IMGDISP to display your

image. The POSITION keyword can easily keep your image in sync
with your map. Plus, it is easy to "zoom" into a map
or image that is set up like this.

The image object associated with our (delayed) Catalyst
Object Library has a built in helper Coordinate object (which
can set up a map projection space if you like). We use this
to "define" the data coordinate space. It is then easy to
interact with the image and find your location in lat/lon
coordinates. It also makes it easy to draw overlays, etc. on
the image because the image knows how to set up its own coordinate
space with respect to the map and to the particular window
it finds itself in.

> While I'm at it, any tips on efficiently reading binary data of
> unknown size into an array - the data consists of 11 values (4byte
> records) for each pixel, but you never know how many pixels there are
> going to be for a particular overpass.
> What I do at the moment is:
>
> ...
>
> openr, lun, file, /get_lun, /swap_if_little_endian
> fileinfo = FStat(lun)
> rows = fileinfo.size/(11*4)
> tmp=fltarr(11,rows)

Isn't this *all* the data? I don't see the need for the
following WHILE loop. I should think you have all of it
right here in one pass! Unless I'm missing something, this
WHILE loop gets executed exactly once.
>
> while not eof(lun) do begin
>   readu,lun,tmp
> endwhile
>
> ....
>
> Then I just need to extract the particular columns I'm interested in
> and make them into a 2D Array for plotting purposes (associated with
> lat/lon) but I'm not sure whether this is the best way to do it.

Yes, this is probably the way to do it. :-)

> - this probably doesn't make much sense so if your interested I could
> email you with some more detailed information.

No, no. No more details unless you also have some cash. :-)

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