Subject: Re: Function Graphics Map Projection Woes
Posted by David Fanning on Mon, 19 Sep 2011 15:33:12 GMT
View Forum Message <> Reply to Message

David Fanning writes:

>
> Folks,
>
> Sigh...
>
> OK, I'm on to my next topic on my list in my effort
> to learn about function graphics: map projections.
>
> Can anyone explain to me why this code only shows
> the Northern hemisphere?
>
>   data = Dist(200)
>   imgObj = Image(data, limit=[-90,-180,90, 180], $
>   grid_units=2, $
>   map_projection='Equirectangular')
>
> Does *anything* in function graphics work correctly!?

Seriously. I am going to absolutely tear my hair out!
Function graphics are so &*^$#ed up they are impossible
to work with!

I have some globally gridded data.

IDL> help, tnmin
TNMIN   FLOAT   = Array[96, 73]

I have *COMPLETELY* regularly spaced longitude and latitude
vectors.

IDL> help, lon,lat
LON   DOUBLE   = Array[96]
LAT   DOUBLE   = Array[73]

IDL> print, lon
     0.00000000   3.7500000   7.5000000   11.250000
15.000000   18.750000   22.500000   26.250000
30.000000...
IDL> print, lat
      90.000000   87.500000   85.000000   82.500000
80.000000   77.500000   75.000000   72.500000
70,000000 ...

I try to display this image in a map projection like this:

```
imgObj = Image(tnmin, lon, lat, limit=[-90,-180,90, 180], $
          grid_units=2, map_projection='Cylindrical')
```

And I get this error:

```
   % IMAGE: X and Y parameters must be evenly spaced
```

What the @*%$!

Cheers,

David
--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.idlcoyote.com/
Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: Function Graphics Map Projection Woes
Posted by David Fanning on Mon, 19 Sep 2011 15:43:28 GMT
View Forum Message <> Reply to Message

David Fanning writes:

```
>  I have *COMPLETELY* regularly spaced longitude and latitude
>  vectors.
>
>  IDL> help, lon,lat
>  LON             DOUBLE    = Array[96]
>  LAT             DOUBLE    = Array[73]
>
>  IDL> print, lon
>       0.00000000       3.7500000       7.5000000       11.250000
>  15.000000       18.750000       22.500000       26.250000
>  30.000000...
>  IDL> print, lat
>        90.000000       87.500000       85.000000       82.500000
>  80.000000       77.500000       75.000000       72.500000
>  70,000000 ...
>
>  I try to display this image in a map projection like this:
>
>  imgObj = Image(tnmin, lon, lat, limit=[-90,-180,90, 180], $
>             grid_units=2, map_projection='Cylindrical')
```

>
> And I get this error:
>
>   % IMAGE: X and Y parameters must be evenly spaced
>
> What the @*%$!

OK, that error message sucks! What it means is that
the vectors have to go from small values to large
values. Sheesh!

This code *finally* give me what I want:

```
  imgObj = Image(tnmin, lon, Reverse(lat), limit=[-90,0,90,360], $
           grid_units=2, map_projection='Cylindrical')
```

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.idlcoyote.com/
Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: Function Graphics Map Projection Woes
Posted by Mark Piper on Mon, 19 Sep 2011 15:48:50 GMT
View Forum Message <> Reply to Message

On 9/19/2011 8:30 AM, David Fanning wrote:
>
> OK, I'm on to my next topic on my list in my effort
> to learn about function graphics: map projections.

By happy coincidence, I'm giving a webinar on using map projections in
IDL, scheduled for October 20. Check back here:

http://www.ittvis.com/language/en-US/EventsTraining/LiveWebS eminars.aspx

a little later today for the announcement. I just finished making my
example programs last week; after I IDLdoc them, I'll post them to
bit.ly/IDL-webinar-files if you'd like to inspect them before the webinar.

> Can anyone explain to me why this code only shows

> the Northern hemisphere?
>
> data = Dist(200)
> imgObj = Image(data, limit=[-90,-180,90, 180], $
> grid_units=2, $
> map_projection='Equirectangular')

You need either the X and Y parameters or the IMAGE_DIMENSIONS and
IMAGE_LOCATIONS keywords to IMAGE.

> Does *anything* in function graphics work correctly!?

I'm standing by my initial assertion that about 95 percent of (New)
Graphics works well. There are bugs (I've logged a bunch) and there are
some routines that we need to improve (e.g., COLORBAR), but with each
IDL 8 release, I've seen NG get better. DG (and therefore CG!) will
always be a part of IDL, but NG provides a nice alternative. I wish I'd
had NG the first time I had to print a plot from IDL.

mp

Subject: Re: Function Graphics Map Projection Woes
Posted by David Fanning on Mon, 19 Sep 2011 15:57:44 GMT
View Forum Message <> Reply to Message

Mark Piper writes:

> By happy coincidence, I'm giving a webinar on using map projections in
> IDL, scheduled for October 20. Check back here:
>
> http://www.ittvis.com/language/en-US/EventsTraining/LiveWebS eminars.aspx
>
> a little later today for the announcement. I just finished making my
> example programs last week; after I IDLdoc them, I'll post them to
> bit.ly/IDL-webinar-files if you'd like to inspect them before the webinar.

Ah, this is exciting news! Unfortunately, I have been
asked to talk about this topic on October 11th, so I
can't wait for enlightenment. :-)

On the other hand, I can use all the help I can get!
I'd be happy to look over the materials ahead of time
and try it with real-world data. Thanks!

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.idlcoyote.com/
Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: Function Graphics Map Projection Woes
Posted by Mark Piper on Mon, 19 Sep 2011 16:09:50 GMT
View Forum Message <> Reply to Message

On 9/19/2011 9:43 AM, David Fanning wrote:
> David Fanning writes:
>
> This code *finally* give me what I want:
>
> imgObj = Image(tnmin, lon, Reverse(lat), limit=[-90,0,90,360], $
> grid_units=2, map_projection='Cylindrical')
>

Here's also a quick example of using the IMAGE_DIMENSIONS and
IMAGE_LOCATION keywords:

```
data = dist(200)
i1 = image(data, $
   limit=[-90,-180,90, 180], $
   grid_units=2, $
   image_dimensions=[360,180], $
   image_location=[-180,-90], $
   margin=0.1, $
   map_projection='Equirectangular', $
   title='Using IMAGE_DIMENSIONS and IMAGE_LOCATION keywords')

mp
```

Subject: Re: Function Graphics Map Projection Woes
Posted by David Fanning on Mon, 19 Sep 2011 16:10:15 GMT
View Forum Message <> Reply to Message

Mark Piper writes:

>> Can anyone explain to me why this code only shows
>> the Northern hemisphere?
>>

>>   data = Dist(200)
>>   imgObj = Image(data, limit=[-90,-180,90, 180], $
>>   grid_units=2, $
>>   map_projection='Equirectangular')
>
> You need either the X and Y parameters or the IMAGE_DIMENSIONS and
> IMAGE_LOCATIONS keywords to IMAGE.

Alright, here is another thing I don't understand,
I guess. This is not as big a problem, now that
I finally got my longitude and latitude vectors
to be accepted, but how should the IMAGE_DIMENSIONS
keyword be used if I *don't* have lon/lat vectors?

In other words, if I set the IMAGE_DIMENSIONS
vector to the actual dimensions of my image, I
find a tiny, little image over in the corner of
my map projection. So, OK, I can set them to something
larger than that, but to what? What is appropriate
in this case? I tried, for example, to set the
Image_Dimensions to [720,360]. This puts the image
on the full map projection, but the image values
are terribly distored. The original image is [97,73].

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.idlcoyote.com/
Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: Function Graphics Map Projection Woes
Posted by David Fanning on Mon, 19 Sep 2011 16:13:00 GMT
View Forum Message <> Reply to Message

Mark Piper writes:

> I'm standing by my initial assertion that about 95 percent of (New)
> Graphics works well

I'm sure this is true. It's just that the first six
things I have tried in function graphics have all

been broken. I calculate the chance of that happening
must be 1 in a million. Go figure! ;-)

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.idlcoyote.com/
Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: Function Graphics Map Projection Woes
Posted by David Fanning on Mon, 19 Sep 2011 20:06:17 GMT
View Forum Message <> Reply to Message

Mark Piper writes:

> By happy coincidence, I'm giving a webinar on using map projections in
> IDL, scheduled for October 20. Check back here:
>
> http://www.ittvis.com/language/en-US/EventsTraining/LiveWebS eminars.aspx
>
> a little later today for the announcement. I just finished making my
> example programs last week; after I IDLdoc them, I'll post them to
> bit.ly/IDL-webinar-files if you'd like to inspect them before the webinar.

OK, so I have some real world map projected gridded data
that I want to see on a map projection. The gridded data
has missing data values. The client wants to see the rest
of the data displayed in 10 separate colors. I've put the
data and the longitude and latitude vectors that go
along with the data here:

http://www.idlcoyote.com/misc/mapdata.sav

Restoring the data file, you will see this:

```
IDL> help, tnmin, lon, lat
TNMIN           FLOAT     = Array[96, 73]
LON            DOUBLE     = Array[96]
LAT            DOUBLE     = Array[73]
```

I can display this data correctly in function graphics with

this code:

```
  ; This data must be reversed in IDL's Y dimension.
  tnmin = Reverse(tnmin, 2)

  ; Missing data is -999.99.
  missing = Where(tnmin EQ -999.99, count)
  IF count GT 0 THEN BEGIN
      tnmin[missing] = !Values.F_NAN
  ENDIF

  ; Scale the data for display
  minvalue = Floor(Min(tnmin, /NAN))
  maxvalue = Ceil(Max(tnmin, /NAN))
  scaledData = BytScl(tnmin, /NAN, TOP=9, $
     MIN=minvalue, MAX=maxvalue)
  IF count GT 0 THEN scaledData[missing] = 255

  ; Display the data.
  LoadCT, 33, NCOLORS=10
  TVLCT, 255, 255, 255, 255 ; Missing values are white.
  TVLCT, rgb, /Get

  w = Window(Dimension=[775, 425])
  imgObj = Image(scaledData, lon, Reverse(lat), $
     limit=[-90,0,90, 356.25], grid_units=2, $
     map_projection='Equirectangular', $
     Position=[0.05,0.05,0.95, 0.80], $
     RGB_TABLE=rgb, /Current)
 c = mapContinents()
 g = Mapgrid(Color='gray')
```

What I cannot do, however, is display a colorbar with just the data colors. This despite having just written the definitive documentation for how to use the Colorbar function with images. :-(

The problem is the white missing value color. It is *always* gettting in the way. I even tried making a "fake" image, as I did last week with the Contour plot, but for some reason I cannot even fathom, this changes the colors on my image plot, even when I "hide" the fake image. :-(

Always interested in new ideas!

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.idlcoyote.com/
Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: Function Graphics Map Projection Woes
Posted by David Fanning on Mon, 19 Sep 2011 22:55:24 GMT
View Forum Message <> Reply to Message

David Fanning writes:

> What I cannot do, however, is display a colorbar with
> just the data colors. This despite having just written
> the definitive documentation for how to use the Colorbar
> function with images. :-(

OK, I have figured out how to create a fake image
that gives me the proper colorbar now.

Now I have some questions about MapGrid.

The spacing of the latitude and longitude lines in my
map grid are too close together. I read the documentation
and I see the keywords GRID_LONGITUDE and GRID_LATITUDE
are to be used to set the "grid spacing in degrees."

So, OK, I define my grid like this:

grid = Mapgrid(Color='gray', Grid_Longitude=60, Grid_Latitude=30)

And *nothing* about the grid changes at all. :-(

Am I reading the documentation wrong, or am I still
on this incredible one in a million losing streak?

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.idlcoyote.com/

Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: Function Graphics Map Projection Woes
Posted by David Fanning on Mon, 19 Sep 2011 23:16:08 GMT
View Forum Message <> Reply to Message

David Fanning writes:

>
> David Fanning writes:
>
>> What I cannot do, however, is display a colorbar with
>> just the data colors. This despite having just written
>> the definitive documentation for how to use the Colorbar
>> function with images. :-(
>
> OK, I have figured out how to create a fake image
> that gives me the proper colorbar now.
>
> Now I have some questions about MapGrid.
>
> The spacing of the latitude and longitude lines in my
> map grid are too close together. I read the documentation
> and I see the keywords GRID_LONGITUDE and GRID_LATITUDE
> are to be used to set the "grid spacing in degrees."
>
> So, OK, I define my grid like this:
>
> grid = Mapgrid(Color='gray', Grid_Longitude=60, Grid_Latitude=30)
>
> And *nothing* about the grid changes at all. :-(
>
> Am I reading the documentation wrong, or am I still
> on this incredible one in a million losing streak?

Also, how do I put a border around the map projection/image?

And, when the image shows up in the display window, it
has this little selection thingy around it that apparently
lets me rotate (I'm not even going to ask why ROTATE
would be the default here!) the image about. How can I
get rid of this selection thing-ama-jig?

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.idlcoyote.com/
Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: Function Graphics Map Projection Woes
Posted by David Fanning on Tue, 20 Sep 2011 15:49:22 GMT
View Forum Message <> Reply to Message

David Fanning writes:

> The spacing of the latitude and longitude lines in my
> map grid are too close together. I read the documentation
> and I see the keywords GRID_LONGITUDE and GRID_LATITUDE
> are to be used to set the "grid spacing in degrees."
>
> So, OK, I define my grid like this:
>
> grid = Mapgrid(Color='gray', Grid_Longitude=60, Grid_Latitude=30)
>
> And *nothing* about the grid changes at all. :-(
>
> Am I reading the documentation wrong, or am I still
> on this incredible one in a million losing streak?

Since no one is helping me here, I've had to resort
to my own (limited) research abilities. :-(

It turns out that this is NOT the way to add a map
grid to an image that is displayed in a map projection.
In direct graphics we think of creating a map projection
on an image, and then "annotating" the map projection with
continental outlines, grids, etc.

This apparently is not the way in function graphics.
At least not for grids. (And maybe not for continental
outlines either. I don't know. The documentation is
really, really bad, as you know. I'm struggling along.)

In any case, I've discovered that to change the spacing
of my latitude and longitude grid lines, I need to
fetch the map projection and grid objects from the image
object, and change THOSE values and give up on creating
the grid object the way I described yesterday. Yesterday,
in fact, I had TWO grid objects, overlaid on one another.

(Although why I didn't see that and/or why those two
keyword didn't do anything that I could see is still a
mystery to me.)

The code should look like this:

```
  w = Window(Dimension=[775, 425])
  imgObj = Image(scaledData, lon, Reverse(lat), $
    limit=[-90,0,90, 356.25], grid_units=2, $
    map_projection='Equirectangular', $
    Position=[0.05,0.05,0.95, 0.80], $
    RGB_TABLE=rgb, /Current)
  map = imgObj.MapProjection
  grid = map.mapgrid
  grid.grid_latitude = 30
  grid.grid_longitude = 60
```

I'm working on an article that will explain what I know
about this topic now. It should be available later today.

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.idlcoyote.com/
Sepore ma de ni thui. ("Perhaps thou speakest truth.")

Subject: Re: Function Graphics Map Projection Woes
Posted by David Fanning on Wed, 21 Sep 2011 14:55:15 GMT
View Forum Message <> Reply to Message

David Fanning writes:

> And, when the image shows up in the display window, it
> has this little selection thingy around it that apparently
> lets me rotate (I'm not even going to ask why ROTATE
> would be the default here!) the image about. How can I
> get rid of this selection thing-ama-jig?

I notice that the selection thingamajig only appears
when I am running the graphics program in a main-level

program. When I run it from a procedure, everything is as I expect it to be.

I think now this is probably not a bug in the function graphics commands. It is probably part of the known bug that keeps main-level programs from working correctly in IDL 8.x.

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.
Coyote's Guide to IDL Programming: http://www.idlcoyote.com/
Sepore ma de ni thui. ("Perhaps thou speakest truth.")