Subject: Map projection of IMAGE() is behaving strangely...
Posted by andeh on Fri, 15 Nov 2013 13:37:52 GMT
View Forum Message <> Reply to Message

Hello,

First time poster, long time reader.

Using IDL 8.2 on linux, I have recently been experimenting with the new IDL IMAGE() function and discovered strange behaviour when projecting an image using the MAP_PROJECTION property (given in the final plot):

```
;; Set up variables
z = BINDGEN(3,5,6) * 21b
lon = FINDGEN(5)*20-60
lat = FINDGEN(6)*20-60


;; Just plot the image.
im0 = IMAGE( z, $
        AXIS_STYLE=2, $
        TITLE='Basic image', $
        LAYOUT=[2,2,1] )

;; Now add coordinates for bottom left point in each pixel.
im1 = IMAGE( z, lon, lat, $
        AXIS_STYLE=2, $
        TITLE='Add x/y coordinates', $
        LAYOUT=[2,2,2], /CURRENT )

;; Let IMAGE() know that we are using lat-lon coordinates.
im2 = IMAGE( z, lon, lat, $
        AXIS_STYLE=2, $
        GRID_UNITS=2, $
        TITLE='+ GRID_UNITS=2', $
        LAYOUT=[2,2,3], /CURRENT )

;; And add a map projection (Sinusoidal in this case, but the same
;; result is seen for Geographic, Orthographic, Interrupted Goode
;; and any other projection I've tried).
im3 = IMAGE( z, lon, lat, $
        AXIS_STYLE=2, $
        GRID_UNITS=2, $
        MAP_PROJECTION='Sinusoidal', $
        TITLE='+ MAP_PROJECTION = Sinusoidal', $
```

LAYOUT=[2,2,4], /CURRENT)

The longitude (x) coordinate appears to be mis-aligned so that the image pixels are stretched in that direction and the image area is filled by z[*,1:-1,*] instead of the full image z[*,*,*]. This doesn't happen in the latitude direction.

Am I missing something fundamental about the MAP_PROJECTION property?

Cheers,

Andy

Subject: Re: Map projection of IMAGE() is behaving strangely...
Posted by David Fanning on Fri, 15 Nov 2013 15:06:11 GMT
View Forum Message <> Reply to Message

Andrew Smith writes:

> The longitude (x) coordinate appears to be mis-aligned so that the image pixels are stretched in that direction and the image area is filled by z[*,1:-1,*] instead of the full image z[*,*,*]. This doesn't happen in the latitude direction.
>
> Am I missing something fundamental about the MAP_PROJECTION property?

It seems to me the weirdness is not from the map projection, which I think might actually be doing the *right* thing. I think the problem is with the Image function. In this call:

```
;; Now add coordinates for bottom left point in each pixel.
im1 = IMAGE( z, lon, lat, $
   AXIS_STYLE=2, $
   TITLE='Add x/y coordinates', $
   LAYOUT=[2,2,2], /CURRENT )
```

The image display shows coordinates that go from -60 to 60 in latitude, and from -60 to 40 in longitude, but this isn't what is in the lat/lon variables:

```
IDL> minmax, lat
MinMax:      -60.0000      40.0000
IDL> minmax, lon
MinMax:      -60.0000      20.0000
```

So, when you get to the map projection, you are getting the wrong
picture of what is suppose to be happening.

Don't know exactly. I just know something *always* seems to be weird
with function graphics routines when you look closely at them. ;-)

Cheers,

David

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

Subject: Re: Map projection of IMAGE() is behaving strangely...
Posted by andeh on Fri, 15 Nov 2013 15:37:05 GMT
View Forum Message <> Reply to Message

On Friday, 15 November 2013 15:06:11 UTC, David Fanning wrote:
> The image display shows coordinates that go from -60 to 60 in latitude,
> and from -60 to 40 in longitude, but this isn't what is in the lat/lon
> variables:
>
> IDL> minmax, lat
> MinMax: -60.0000   40.0000
> IDL> minmax, lon
> MinMax: -60.0000   20.0000
>

Hi David,

Thanks for your response. I guess this means that I am incorrect that the lat/lon values represent the bottom-left point in each pixel once we get to the map projection. I expected the upper range of the boxes to be 20 degrees larger than the maximum values as that was my bin width in the image.

Just to check, my output looks like this: http://goo.gl/1aY3ab
Did yours?

Moving the first column of data to the left of 60W line seems like a very strange feature of the

projection... Especially since the xrange and all aspects of the y plotting behave as I was expecting.

Cheers,
Andy

Subject: Re: Map projection of IMAGE() is behaving strangely...
Posted by andeh on Fri, 15 Nov 2013 15:43:33 GMT
View Forum Message <> Reply to Message

Woops!

I just noticed the bottom Y value is missing as well! Disaster.

Subject: Re: Map projection of IMAGE() is behaving strangely...
Posted by David Fanning on Fri, 15 Nov 2013 15:52:01 GMT
View Forum Message <> Reply to Message

AJAS writes:

> Thanks for your response. I guess this means that I am incorrect that the lat/lon values represent the bottom-left point in each pixel once we get to the map projection. I expected the upper range of the boxes to be 20 degrees larger than the maximum values as that was my bin width in the image.

Well, hard to say without more investigation. I'm never particularly optimistic when it comes to IDL map projections, though. They always seem to be doing something different from what I *think* they should be doing. :-)

> Just to check, my output looks like this: http://goo.gl/1aY3ab
> Did yours?

More or less, yes, but I'm still not absolutely convinced the problem is in the map projection part of the code. In any case, have to run. One of the kids has strep and has no "soft" food in the house. Running to the rescue now. :-)

Cheers,

David
--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.

Coyote's Guide to IDL Programming: http://www.idlcoyote.com/
Sepore ma de ni thue. ("Perhaps thou speakest truth.")

Subject: Re: Map projection of IMAGE() is behaving strangely...
Posted by andeh on Fri, 15 Nov 2013 15:59:33 GMT
View Forum Message <> Reply to Message

Good luck with that! Thanks again. A

Subject: Re: Map projection of IMAGE() is behaving strangely...
Posted by Andy Sayer on Fri, 15 Nov 2013 16:35:05 GMT
View Forum Message <> Reply to Message

On Friday, November 15, 2013 10:59:33 AM UTC-5, AJAS wrote:
> Good luck with that! Thanks again. A

Andy, I'm going to email you something which may help (uses IDL 7 code though).

Andy

Subject: Re: Map projection of IMAGE() is behaving strangely...
Posted by MP on Fri, 15 Nov 2013 16:49:07 GMT
View Forum Message <> Reply to Message

On Friday, November 15, 2013 6:37:52 AM UTC-7, AJAS wrote:
>
> ;; And add a map projection (Sinusoidal in this case, but the same
> ;; result is seen for Geographic, Orthographic, Interrupted Goode
> ;; and any other projection I've tried).
> im3 = IMAGE(z, lon, lat, $
> AXIS_STYLE=2, $
> GRID_UNITS=2, $
> MAP_PROJECTION='Sinusoidal', $
> TITLE='+ MAP_PROJECTION = Sinusoidal', $
> LAYOUT=[2,2,4], /CURRENT)
>

This is an aside, and may not be useful in this case, but I strongly recommend using MAP, then IMAGE with /OVERPLOT. Although they should behave identically, I've had more success with this technique than using the MAP_PROJECTION keyword to IMAGE.

mp

Subject: Re: Map projection of IMAGE() is behaving strangely...
Posted by brainwasher on Fri, 15 Nov 2013 16:52:07 GMT
View Forum Message <> Reply to Message

On Friday, 15 November 2013 16:43:33 UTC+1, AJAS wrote:
> Woops!
>
>
>
> I just noticed the bottom Y value is missing as well! Disaster.

Well, that problem in not only related to IMAGE function. It is found also in iTools and function iImage, and it is inherited from who knows where and when. I was really expecting that new graphics will solve it, but when i tried it for the first time I was dissapointed.

I was always using a quite straightforward way around, to give to idl one row and column it can swallow:

```
z_enlarged = FLTARR(3,nlon+1, nlat+1)
z_enlarged[0,1,1] = z
lon_enlarged = ((max(lon)-min(lon))/float(nlon))*findgen(nlon+1)+min(lon)
lat_enlarged = ((max(lat)-min(lat))/float(nlat))*findgen(nlat+1)+min(lat)
```

It is not perfect still, because idl shrinks the field little bit, but with a large number of points it looks good.

Any other solution?

Subject: Re: Map projection of IMAGE() is behaving strangely...
Posted by David Fanning on Fri, 15 Nov 2013 17:06:37 GMT
View Forum Message <> Reply to Message

brainwasher@gmail.com writes:

> I was always using a quite straightforward way around, to give to idl one row and column it can swallow:

Love this solution! My favorite moment in an IDL class is when I
explain how something works and the whole class stares at me with their
mouths open. ;-)

Cheers,

David
--
David Fanning, Ph.D.
Fanning Software Consulting, Inc.

Subject: Re: Map projection of IMAGE() is behaving strangely...
Posted by Andy Sayer on Fri, 15 Nov 2013 17:43:52 GMT
View Forum Message <> Reply to Message

Brain, I wonder if that could be related at all to the odd behaviour (reported to ExelisVis but not something they're going to fix) of map_image truncating edges?
https://groups.google.com/forum/#!searchin/comp.lang.idl-pvw
ave/map_image/comp.lang.idl-pvwave/3BmVLThB8Lk/_QUaZ--wehkJ

On Friday, November 15, 2013 11:52:07 AM UTC-5, brain...@gmail.com wrote:
> On Friday, 15 November 2013 16:43:33 UTC+1, AJAS wrote:
>
>> Woops!
>
>>
>
>>
>
>>
>
>> I just noticed the bottom Y value is missing as well! Disaster.
>
>
>
> Well, that problem in not only related to IMAGE function. It is found also in iTools and function iImage, and it is inherited from who knows where and when. I was really expecting that new graphics will solve it, but when i tried it for the first time I was dissapointed.
>
>
>
> I was always using a quite straightforward way around, to give to idl one row and column it can swallow:
>
>
>
> z_enlarged = FLTARR(3,nlon+1, nlat+1)
>
> z_enlarged[0,1,1] = z
>
> lon_enlarged = ((max(lon)-min(lon))/float(nlon))*findgen(nlon+1)+min(lon)
>
> lat_enlarged = ((max(lat)-min(lat))/float(nlat))*findgen(nlat+1)+min(lat)
>
>

>
> It is not perfect still, because idl shrinks the field little bit, but with a large number of points it looks good.
>
>
>
> Any other solution?

Subject: Re: Map projection of IMAGE() is behaving strangely...
Posted by Brian McNoldy on Tue, 17 Oct 2017 18:33:43 GMT
View Forum Message <> Reply to Message

I see this thread is nearly 4 years old now, but it just came in handy for me this week. I was trying to plot an image with a map projection and then overlay filled continents... all using functional graphics. It was frustrating, but I ended up using a combination of two tricks suggested in here:

1) I needed to add a blank row to the beginning of my image array because the map projection process would "eat"/truncate the first row (but not column). Doing so magically solved that quirk. No idea what's going on there.

2) I called MAP first with my projection of choice, and then added the IMAGE with /OVERPLOT.

The guts of my successful result looks like:
```
 m=map('Mollweide',center_longitude=clon,limit=limit,position =position,$
dimensions=[1200,500])

 i=image(array,lons[0:-2],lats,grid_units='degrees',rgb_table =table,$
image_dimensions=[360,90],image_location=[0,-45],/overplot)
```

(My data array spanned all longitudes but only -45 to +45 in latitude.)
(The different treatment of the lons and lats arrays was because of needing to extend the array and latitude vector as described in (1)).

It feels like a hack, but the result is what I was hoping for.

FYI,
Brian

On Friday, November 15, 2013 at 12:43:52 PM UTC-5, AMS wrote:
> Brain, I wonder if that could be related at all to the odd behaviour (reported to ExelisVis but not something they're going to fix) of map_image truncating edges?
https://groups.google.com/forum/#!searchin/comp.lang.idl-pvw
ave/map_image/comp.lang.idl-pvwave/3BmVLThB8Lk/_QUaZ--wehkJ
>
> On Friday, November 15, 2013 11:52:07 AM UTC-5, brain...@gmail.com wrote:
>> On Friday, 15 November 2013 16:43:33 UTC+1, AJAS wrote:

>>
>>> Woops!
>>
>>>
>>
>>>
>>
>>>
>>
>>> I just noticed the bottom Y value is missing as well! Disaster.
>>
>>
>>
>> Well, that problem in not only related to IMAGE function. It is found also in iTools and function iImage, and it is inherited from who knows where and when. I was really expecting that new graphics will solve it, but when i tried it for the first time I was dissapointed.
>>
>>
>>
>> I was always using a quite straightforward way around, to give to idl one row and column it can swallow:
>>
>>
>>
>> z_enlarged = FLTARR(3,nlon+1, nlat+1)
>>
>> z_enlarged[0,1,1] = z
>>
>> lon_enlarged = ((max(lon)-min(lon))/float(nlon))*findgen(nlon+1)+min(lon)
>>
>> lat_enlarged = ((max(lat)-min(lat))/float(nlat))*findgen(nlat+1)+min(lat)
>>
>>
>>
>> It is not perfect still, because idl shrinks the field little bit, but with a large number of points it looks good.
>>
>>
>>
>> Any other solution?