
Subject: Re: map_image and latmin/latmax, lonmin/lonmax - edges or centres?

Posted by [David Fanning](#) on Wed, 25 Sep 2013 19:18:54 GMT

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

AMS writes:

> I'm trying to map a regularly-gridded image (in IDL 7.1.1). I'm using map_image to figure out where in the output window the image should go. But, it's not lining up exactly as I expect with e.g. coastlines. I am using compress=1 (i.e. inverse transform each point), so I don't think it is that. I wonder if I am using the latmin/latmax and lonmin/lonmax keywords incorrectly.

>

> The documentation (http://www.exelisvis.com/docs/MAP_IMAGE.html) says this:

>

> LATMIN

> The latitude corresponding to the first row of Image. The default is -90 degrees. Note also that $-90^\circ = \text{LATMIN} < \text{LATMAX} = 90^\circ$.

> LATMAX

> The latitude corresponding to the last row of Image. The default value is 90 degrees. Note also that $-90^\circ = \text{LATMIN} < \text{LATMAX} = 90^\circ$.

> LONMIN

> The longitude corresponding to the first (leftmost) column of the Image argument. Select LONMIN so that $-180^\circ = \text{LONMIN} = 180^\circ$. The default value is -180.

> LONMAX

> The longitude corresponding to the last (rightmost) column of the Image argument. Select LONMAX so that it is larger than LONMIN. If the longitude of the last column is equal to $(\text{LONMIN} - (360. / \text{Nx})) \text{ MODULO } 360$, it is assumed that the image covers all longitudes (Nx being the total number of columns in the Image argument).

>

> My question: are these keywords expecting you pass the centre values for the edge rows/columns? Lower-left corners? Or the outer values (lower-left and upper-right)? (The difference being half the pixel size in lat/lon.)

>

> My testing suggests that using the outer values of these corner pixels is probably the intention (in that the image then covers the intended portion of the output window), but to me at least the wording in the documentation could be clearer. And the end result does not look entirely correct to me, as it looks like some pixels don't match up entirely with the coastlines (from map_continents,/hires,/coast), although I don't know whether that is something that I am doing incorrectly, a deficiency with the dataset I am using (it isn't one created here), or a deficiency in the IDL coastline database.

>

> If someone in the know could confirm the intended use of these map_image keywords, I'd appreciate it. :)

It seems to me the answer to your questions is "none of the above". I think the only way these keywords make any sense at all is if they represent the limits of the image rectangle. So, I would say, they should

be the outside edges of the pixels in the very center of the image, as if you set up orthogonal axes with 0,0 at the center of the image. These values will represent the values where the axes leave the image.

Of course, none of this makes much sense in lat/lon space, so I presume the program converts to xy projected meter space (where all this *does* make sense) first, then does the warping into this space. But, that's just my guess. I don't often use Map_Image.

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_image and latmin/latmax, lonmin/lonmax - edges or centres?

Posted by [Andy Sayer](#) on Wed, 25 Sep 2013 20:19:13 GMT

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

After fiddling some more, I agree that it's probably the outer coordinates of the image rectangle that it wants. However, I'm still encountering some odd behaviour. For example, if I take regular 1 degree gridded data and try to map that, things like the US-Canada border (at 49 degrees N) don't lie exactly on grid cell borders (as they should) but are slightly shifted. I'm going to leave it for the rest of the day in case my overtired mind is overlooking something, but if I can't figure it out tomorrow, will create some code which reproduces the issue and post it.

Adny

On Wednesday, September 25, 2013 3:18:54 PM UTC-4, David Fanning wrote:

> AMS writes:

>

>

>

>> I'm trying to map a regularly-gridded image (in IDL 7.1.1). I'm using

>

> map_image to figure out where in the output window the image should go.

>

> But, it's not lining up exactly as I expect with e.g. coastlines. I am

>

> using compress=1 (i.e. inverse transform each point), so I don't think

>

> it is that. I wonder if I am using the latmin/latmax and lonmin/lonmax

```

>
> keywords incorrectly.
>
>>
>
>> The documentation ( http://www.exelisvis.com/docs/MAP\_IMAGE.html ) says this:
>
>>
>
>> LATMIN
>
>> The latitude corresponding to the first row of Image. The default is -90 degrees. Note also that
-90°= LATMIN < LATMAX = 90°.
>
>> LATMAX
>
>> The latitude corresponding to the last row of Image. The default value is 90 degrees. Note
also that -90°= LATMIN < LATMAX = 90°.
>
>> LONMIN
>
>> The longitude corresponding to the first (leftmost) column of the Image argument. Select
LONMIN so that -180°= LONMIN = 180°. The default value is -180.
>
>> LONMAX
>
>> The longitude corresponding to the last (rightmost) column of the Image argument. Select
LONMAX so that it is larger than LONMIN. If the longitude of the last column is equal to (LONMIN
- (360. /Nx)) MODULO 360, it is assumed that the image covers all longitudes (Nx being the total
number of columns in the Image argument).
>
>>
>
>> My question: are these keywords expecting you pass the centre values for the edge
rows/columns? Lower-left corners? Or the outer values (lower-left and upper-right)? (The
difference being half the pixel size in lat/lon.)
>
>>
>
>> My testing suggests that using the outer values of these corner pixels is probably the intention
(in that the image then covers the intended portion of the output window), but to me at least the
wording in the documentation could be clearer. And the end result does not look entirely correct to
me, as it looks like some pixels don't match up entirely with the coastlines (from
map_continents,/hires,/coast), although I don't know whether that is something that I am doing
>
> incorrectly, a deficiency with the dataset I am using (it isn't one created here), or a deficiency in
the IDL coastline database.
>

```

>>
>
>> If someone in the know could confirm the intended use of these map_image keywords, I'd appreciate it. :)
>
>
>
> It seems to me the answer to your questions is "none of the above". I
>
> think the only way these keywords make any sense at all is if they
>
> represent the limits of the image rectangle. So, I would say, they should
>
> be the outside edges of the pixels in the very center of the image, as
>
> if you set up orthogonal axes with 0,0 at the center of the image. These
>
> values will represent the values where the axes leave the image.
>
>
>
> Of course, none of this makes much sense in lat/lon space, so I presume
>
> the program converts to xy projected meter space (where all this *does*
>
> make sense) first, then does the warping into this space. But, that's
>
> just my guess. I don't often use Map_Image.
>
>
>
> 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_image and latmin/latmax, lonmin/lonmax - edges or centres?
Posted by [David Fanning](#) on Wed, 25 Sep 2013 20:35:01 GMT
[View Forum Message](#) <> [Reply to Message](#)

AMS writes:

> After fiddling some more, I agree that it's probably the outer coordinates of the image rectangle that it wants. However, I'm still encountering some odd behaviour. For example, if I take regular 1 degree gridded data and try to map that, things like the US-Canada border (at 49 degrees N) don't lie exactly on grid cell borders (as they should) but are slightly shifted. I'm going to leave it for the rest of the day in case my overtired mind is overlooking something, but if I can't figure it out tomorrow, will create some code which reproduces the issue and post it.

You might find that the GSHHS database is more accurate than the database that comes with IDL:

http://www.idlcoyote.com/map_tips/gshhs.html

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_image and latmin/latmax, lonmin/lonmax - edges or centres?
Posted by [Andy Sayer](#) on Thu, 26 Sep 2013 12:50:59 GMT
[View Forum Message](#) <> [Reply to Message](#)

I keep meaning to rewrite my mapping routines to include GSHHS but didn't get round to it yet... :)

I think in this case it isn't the IDL borders & boundaries database which is in the wrong, but rather my use of map_image. The reason is that if I draw the data in my image pixel-by-pixel using polyfill it lines up exactly correctly, and it's only when I use map_image that the data is offset from where it should be. So something must be up with these offsets.

Andy

On Wednesday, September 25, 2013 4:35:01 PM UTC-4, David Fanning wrote:

> AMS writes:

>
>
>

>> After fiddling some more, I agree that it's probably the outer coordinates of the image rectangle that it wants. However, I'm still encountering some odd behaviour. For example, if I take regular 1 degree gridded data and try to map that, things like the US-Canada border (at 49 degrees N) don't lie exactly on grid cell borders (as they should) but are slightly shifted. I'm going to leave it for the rest of the day in case my overtired mind is overlooking something, but

>

> if I can't figure it out tomorrow, will create some code which reproduces the issue and post it.

>
>
>

> You might find that the GSHHS database is more accurate than the

>

> database that comes with IDL:

>
>
>

> http://www.idlcoyote.com/map_tips/gshhs.html

>
>
>

> 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_image and latmin/latmax, lonmin/lonmax - edges or centres?

Posted by [Andy Sayer](#) on Thu, 26 Sep 2013 13:17:40 GMT

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

Ok, here is some code which will reproduce the problem (IDL 7.1.1). This should draw an image in a 5x5 degree box from 47-52 N, 115-110 W. It puts the image in the right place, but the edge columns and rows are chopped in half in size (they should all be the same size). So you can see that this means all the points are offset from where they really should be. Does anyone know what is happening here?

```
; Define some dummy data
datavals=intarr(5,5)
for i=0,4 do begin
  for j=0,4 do begin
    datavals[i,j]=i+j
  endfor
endfor

; Scale the data to byte range 0=255
scaled_data=1+254*(datavals)/max(datavals)

; Load a colour table
loadct,3

; Set up a map projection covering the region of interest and a bit more
window,0
map_set,latdel=1,londel=1,limit=[45,-118,55,-108],/noborder, xmargin=[2,2],ymargin=[2,2]

; Scale image to data: assume the 5x5 grid is a dataset at 1 degree resolution.
; So put the box from, say, 47-52 N, 115 to 110 W

cola = map_image(scaled_data,startx,starty,xsize,ysize,compress=1,$
                 latmin=47.0,latmax=52.0,$
                 lonmin=-115.,lonmax=-110.)

; Put the image on the map
tv,cola,startx,starty,xsize=ysize

; Add borders
map_continents,/usa

end
```

On Thursday, September 26, 2013 8:50:59 AM UTC-4, AMS wrote:

> I keep meaning to rewrite my mapping routines to include GSHHS but didn't get round to it yet... :)

>

>

>

> I think in this case it isn't the IDL borders & boundaries database which is in the wrong, but

rather my use of map_image. The reason is that if I draw the data in my image pixel-by-pixel using polyfill it lines up exactly correctly, and it's only when I use map_image that the data is offset from where it should be. So something must be up with these offsets.

```
>
>
>
> Andy
>
>
>
> On Wednesday, September 25, 2013 4:35:01 PM UTC-4, David Fanning wrote:
>
>> AMS writes:
>
>>
>
>>
>
>>
>
>>> After fiddling some more, I agree that it's probably the outer coordinates of the image
rectangle that it wants. However, I'm still encountering some odd behaviour. For example, if I take
regular 1 degree gridded data and try to map that, things like the US-Canada border (at 49
degrees N) don't lie exactly on grid cell borders (as they should) but are slightly shifted. I'm going
to leave it for the rest of the day in case my overtired mind is overlooking something, but
>
>>
>
>> if I can't figure it out tomorrow, will create some code which reproduces the issue and post it.
>
>>
>
>>
>
>>
>
>> You might find that the GSHHS database is more accurate than the
>
>>
>
>> database that comes with IDL:
>
>>
>
>>
>
>>
```



```
>> http://www.idlcoyote.com/map_tips/gshhs.html
>
>>
>
>>
>
>>
>
>> 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_image and latmin/latmax, lonmin/lonmax - edges or centres?
Posted by [David Fanning](#) on Thu, 26 Sep 2013 14:39:29 GMT
[View Forum Message](#) <> [Reply to Message](#)

AMS writes:

> Ok, here is some code which will reproduce the problem (IDL 7.1.1). This should draw an image in a 5x5 degree box from 47-52 N, 115-110 W. It puts the image in the right place, but the edge columns and rows are chopped in half in size (they should all be the same size). So you can see that this means all the points are offset from where they really should be. Does anyone know what is happening here?

My guess is this is just an artifact of using the Map_Set way of doing things, which is very, very old and not of professional grade in terms of map projection software. Using the Map_Proj_*** routines, however, does result in what I think you expect. I added this code to the end of yours to give you the same perspective.

```
mapStruct = Map_Proj_Init(117,Limit=[45,-118,55,-108], $
    center_lon=-112.5)
cola1 = Map_Proj_Image(scaled_data, [-115, 47, -110, 52], $
    Dimensions=[xsize, ysize], Map_Struct=mapStruct)
window, 1
map_set,latdel=1,lonel=1,limit=[45,-118,55,-108],/noborder, $
    xmargin=[2,2],ymargin=[2,2]
tv, cola1, startx,starty,xsize=xsize,ysize=ysize
```

Of course, you have to get your head around the LIMIT being expressed in a different order in the Map_Proj_Init and Map_Proj_Image routines, but once you are burned 8-10 times it becomes second nature to you. :-)

I would encourage everyone to use the Map_Proj_**** routines for map projection work. They are old, but at least professional quality. Of course, they are hard to work with, which is why I generally work with the Coyote Library map projection routines instead, which are wrappers for the Map_Proj_*** routines and make this kind of thing much easier.

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_image and latmin/latmax, lonmin/lonmax - edges or centres?
Posted by [Paul Van Delst\[1\]](#) on Thu, 26 Sep 2013 16:04:45 GMT
[View Forum Message](#) <> [Reply to Message](#)

On 09/26/13 10:39, David Fanning wrote:

> once you are burned 8-10 times it becomes second nature to you. :-)

You funny.

cheers,

paulv

Subject: Re: map_image and latmin/latmax, lonmin/lonmax - edges or centres?
Posted by [Andy Sayer](#) on Thu, 26 Sep 2013 16:18:08 GMT
[View Forum Message](#) <> [Reply to Message](#)

Hi David,

Almost, but not quite. Now, all the grid cells are the same size. But, it only puts 4x4 cells in there (instead of 5x5). It looks like the left column and bottom row are being cut off the image correctly.

Andy

On Thursday, September 26, 2013 10:39:29 AM UTC-4, David Fanning wrote:

> AMS writes:

>

>

>

>> Ok, here is some code which will reproduce the problem (IDL 7.1.1). This should draw an image in a 5x5 degree box from 47-52 N, 115-110 W. It puts the image in the right place, but the edge columns and rows are chopped in half in size (they should all be the same size). So you can see that this means all the points are offset from where they really should be. Does anyone know what is happening here?

>

>

>

> My guess is this is just an artifact of using the Map_Set way of doing

>

> things, which is very, very old and not of professional grade in terms

>

> of map projection software. Using the Map_Proj_*** routines, however,

>

> does result in what I think you expect. I added this code to the end

>

> of yours to give you the same perspective.

>

```
>
>
> mapStruct = Map_Proj_Init(117,Limit=[45,-118,55,-108], $
>   center_lon=-112.5)
>
> cola1 = Map_Proj_Image(scaled_data, [-115, 47, -110, 52], $
>   Dimensions=[xsize, ysize], Map_Struct=mapStruct)
>
> window, 1
>
> map_set,latdel=1,londel=1,limit=[45,-118,55,-108],/noborder, $
>   xmargin=[2,2],ymargin=[2,2]
>
> tv, cola1, startx,starty,xsize=xsize,ysize=ysize
>
>
>
> Of course, you have to get your head around the LIMIT being expressed in
>
> a different order in the Map_Proj_Init and Map_Proj_Image routines, but
>
> once you are burned 8-10 times it becomes second nature to you. :-)
```

>

>

> I would encourage everyone to use the Map_Proj_**** routines for map

>

> projection work. They are old, but at least professional quality. Of

>

> course, they are hard to work with, which is why I generally work with

>

> the Coyote Library map projection routines instead, which are wrappers

>

> for the Map_Proj_*** routines and make this kind of thing much easier.

>

>

>

> 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_image and latmin/latmax, lonmin/lonmax - edges or centres?
Posted by [David Fanning](#) on Thu, 26 Sep 2013 16:49:26 GMT
[View Forum Message](#) <> [Reply to Message](#)

AMS writes:

> Almost, but not quite. Now, all the grid cells are the same size. But, it only puts 4x4 cells in there (instead of 5x5). It looks like the left column and bottom row are being cut off the image correctly.

I have to wave my hands a little bit here, but let me try to explain why I think this might be the correct result.

We have five pixels covering a range of, say, 47 to 52 degrees. If we assume the first pixel value represents the bottom edge of the image and the fifth pixel value represents the top edge of the image, then, in fact, we have only four values that can go into that range. Remember, we are interpolating the data into this space. Think of a vertical color bar. How many horizontal lines would you have to draw to separate four colors. The answer is five. One at the bottom, three to separate colors, and one at the top.

I think that is what is happening here. The five values of our image are analogous to the locations of five vertical lines that we are using to "separate" the colors. We can only separate them into four colors with five lines. So, I think this is just the way interpolation works.

This example is a bit contrived, of course. With these kinds of map projections and this image there would be no need to do any interpolation at all. But, in general, I think the interpolation is being done correctly by Map_Proj_Image.

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_image and latmin/latmax, lonmin/lonmax - edges or centres?
Posted by [Andy Sayer](#) on Thu, 26 Sep 2013 17:00:10 GMT
[View Forum Message](#) <> [Reply to Message](#)

I agree with your explanation for this behaviour, but the question is then how to map the full image. ;-) There must be a way to do this which isn't excessively complicated, as I don't see that it is a very unusual problem.

It seems that with map_image I need to reposition to use corner grid cell centres rather than edges, and then I end up losing half a grid cell from each edge. And with map_proj_init, to map the whole thing I'd need to insert a dummy row or column on one edge.

I opened a support ticket with Exelis but didn't hear anything back yet.

Andy

On Thursday, September 26, 2013 12:49:26 PM UTC-4, David Fanning wrote:

> AMS writes:

>
>
>
>

>> Almost, but not quite. Now, all the grid cells are the same size. But, it only puts 4x4 cells in there (instead of 5x5). It looks like the left column and bottom row are being cut off the image correctly.

>
>
>

> I have to wave my hands a little bit here, but let me try to explain why

>

> I think this might be the correct result.

>
>
>

> We have five pixels covering a range of, say, 47 to 52 degrees. If we

>

> assume the first pixel value represents the bottom edge of the image and

>

> the fifth pixel value represents the top edge of the image, then, in

>

> fact, we have only four values that can go into that range. Remember, we

>

> are interpolating the data into this space. Think of a vertical color

>
> bar. How many horizontal lines would you have to draw to separate four
>
> colors. The answer is five. One at the bottom, three to separate colors,
>
> and one at the top.
>
>
>
> I think that is what is happening here. The five values of our image are
>
> analogous to the locations of five vertical lines that we are using to
>
> "separate" the colors. We can only separate them into four colors with
>
> five lines. So, I think this is just the way interpolation works.
>
>
>
> This example is a bit contrived, of course. With these kinds of map
>
> projections and this image there would be no need to do any
>
> interpolation at all. But, in general, I think the interpolation is
>
> being done correctly by Map_Proj_Image.
>
>
>
> 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_image and latmin/latmax, lonmin/lonmax - edges or centres?
Posted by [David Fanning](#) on Thu, 26 Sep 2013 17:08:00 GMT
[View Forum Message](#) <> [Reply to Message](#)

David Fanning writes:

> I think that is what is happening here. The five values of our image are
> analogous to the locations of five vertical lines that we are using to
> "separate" the colors. We can only separate them into four colors with
> five lines. So, I think this is just the way interpolation works.

This is one reason, of course, why most people who work with images and map projections would NEVER think of using Map_Proj_Image or Map_Image, either. Interpolation fools around with your very hard to acquire and expensive image pixel values.

It makes MUCH more sense to associate a map projection with the image itself. In other words, leave the image alone, and just associate the map projection with the image. This allows you to annotate the image with map information without actually changing the shape or pixel values of the image itself.

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_image and latmin/latmax, lonmin/lonmax - edges or centres?
Posted by [David Fanning](#) on Thu, 26 Sep 2013 17:10:50 GMT
[View Forum Message](#) <> [Reply to Message](#)

AMS writes:

> I opened a support ticket with Exelis but didn't hear anything back yet.

It is not clear to me that the support people at Exelis understand map projections any better than the rest of us. :-)

Cheers,

David

--

David Fanning, Ph.D.

Subject: Re: map_image and latmin/latmax, lonmin/lonmax - edges or centres?
Posted by [Andy Sayer](#) on Thu, 26 Sep 2013 17:11:15 GMT

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

Right, I agree in principle, but sometimes you want to take images from different sources and warp them to a common map projection to make a direct comparison. And if there is a map_image function in IDL, one would hope that it would work in a manner similar to intuition and/or the documentation.

Andy

On Thursday, September 26, 2013 1:08:00 PM UTC-4, David Fanning wrote:

> David Fanning writes:

>
>
>

>> I think that is what is happening here. The five values of our image are

>

>> analogous to the locations of five vertical lines that we are using to

>

>> "separate" the colors. We can only separate them into four colors with

>

>> five lines. So, I think this is just the way interpolation works.

>

>

>

> This is one reason, of course, why most people who work with images and

>

> map projections would NEVER think of using Map_Proj_Image or Map_Image,

>

> either. Interpolation fools around with your very hard to acquire and

>

> expensive image pixel values.

>

>

>

> It makes MUCH more sense to associate a map projection with the image

>

> itself. In other words, leave the image alone, and just associate the

>

> map projection with the image. This allows you to annotate the image

>

> with map information without actually changing the shape or pixel values

>
> of the image itself.
>
>
>
> 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_image and latmin/latmax, lonmin/lonmax - edges or centres?
Posted by [David Fanning](#) on Thu, 26 Sep 2013 17:16:35 GMT
[View Forum Message](#) <> [Reply to Message](#)

AMS writes:

> Right, I agree in principle, but sometimes you want to take images from different sources and warp them to a common map projection to make a direct comparison. And if there is a map_image function in IDL, one would hope that it would work in a manner similar to intuition and/or the documentation.

Ah, not a computer scientist, I see. ;-)

Yes, we all have hopes and dreams that are sometimes dashed on the shoals of reality. "Get used to disappointment," is still the best advice I can offer.

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_image and latmin/latmax, lonmin/lonmax - edges or centres?
Posted by [David Fanning](#) on Thu, 26 Sep 2013 17:52:40 GMT
[View Forum Message](#) <> [Reply to Message](#)

AMS writes:

> I am still young/naive enough that my enthusiasm, while not limitless, has not yet entirely
burned away to leave the cold ashes of cynicism. ;-)

I hear you, man. And envy you. ;-)

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_image and latmin/latmax, lonmin/lonmax - edges or centres?
Posted by [Andy Sayer](#) on Fri, 27 Sep 2013 20:35:37 GMT
[View Forum Message](#) <> [Reply to Message](#)

I heard back from Exelis tech support.

It turns out that map_image does expect the CENTRE, not outer edge, coordinates of the edge rows/columns. If you do this, things line up, but then you still end up with a half-pixel empty border around your image (as the edge pixels are still truncated). I've requested the documentation be updated so that this (to my mind) ambiguous wording and counterintuitive behaviour is documented more clearly.

The suggestion that was made to me was to enlarge my array using congrid(), so that when my image is plotting, the missing half-pixel border will look smaller. While this may be a workaround from a practical point of view, it feels very kludgy, and I don't think it is entirely satisfactory behaviour for several reasons. So I have also requested that this be considered as a bug in map_image, and either fixed, or else add a keyword so that the routine does what would naively be expected.

We'll see what happens...

Andy

On Thursday, September 26, 2013 1:52:40 PM UTC-4, David Fanning wrote:

> AMS writes:

>

>

>

>> I am still young/naive enough that my enthusiasm, while not limitless, has not yet entirely burned away to leave the cold ashes of cynicism. ;-)

>

>

>

> I hear you, man. And envy you. ;-)

>

>

>

> 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_image and latmin/latmax, lonmin/lonmax - edges or centres?

Posted by [David Fanning](#) on Fri, 27 Sep 2013 21:09:01 GMT

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

AMS writes:

> The suggestion that was made to me was to enlarge my array using `congrid()`, so that when my image is plotting, the missing half-pixel border will look smaller. While this may be a workaround from a practical point of view, it feels very kludgy, and I don't think it is entirely satisfactory behaviour for several reasons. So I have also requested that this be considered as a bug in `map_image`, and either fixed, or else add a keyword so that the routine does what would naively be expected.

Yes, this recalls a conversation of my own, in which I pointed out to them that `NLEVELS=12` on the `Contour` command was only producing 9 contour levels. They suggested I use `NLEVELS=14` or `NLEVELS=15`, then. Well, yes, but that kind of "solution" doesn't really inspire a lot confidence in the software, does it.

This, of course, eventually lead to the entire Coyote Library, just because I couldn't stand relaying this incredibly awkward solution to people attending IDL programming classes. And, once I got started I just thought, "Oh, what the hell, as long as I'm fixing this I might as well fix everything that bugs me." :-)

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.")
