Subject: Re: Display and Navigate Image in IDL 8.2
Posted by David Fanning on Fri, 07 Sep 2012 16:02:43 GMT
View Forum Message <> Reply to Message

alx writes:

> Yes, I guess that you can !
> If you got an image from Google which is properly rectified, you do not need for any further map
projection, as far as the axis labeling is in meter/kilometer.
> If now you want axes graduated in longitude/latitude, you only need for a projection tool in order
to manage the scaling which is (slightly in your case) not linear.
> I suppose that 'map_proj_init'/'map_proj_inverse' is what you can use. Then you can add axes
with the proper labeling. This is likely what MAP_PROJECTION is doing for you in the IMAGE
function! Is'nt it ?
> Maybe what I say is pure non-sense, because I am not a specialist of mapping.
> I am just a scientist who *needs* for clever and efficient programming tools, what IDL still is, I
guess, in spite of some irritating and uncorrected lack for a serious (i.e. usable) documentation.

You can see the problem with your particular approach
by using a larger area:

```
googleStr = "http://maps.googleapis.com/maps/api/staticmap?" + $
   "center=40.6000,-105.1000&zoom=4&size=600x600" + $
   "&maptype=terrain&sensor=false&format=png32"
netObject = Obj_New('IDLnetURL')
void = netObject -> Get(URL=googleStr, FILENAME="googleimg.png")
Obj_Destroy, netObject
googleImage = Read_Image('googleimg.png')


centerlat = 40.6000d
centerlon = -105.1000d
res = cgGoogle_MetersPerPixel(4)
map = Map_Proj_Init('mercator', /gctp, ELLIPSOID='wgs 84')
cm = map_proj_forward(centerlon, centerlat, MAP=map)
xrange = cm[0] + [-300,300]*res
yrange = cm[1] + [-300,300]*res
xr = map_proj_inverse(xrange, yrange, MAP=map)
obj = IMAGE(googleimage, $
   GRID_UNITS=2, $    ;units of degrees
   MARGIN=0.1, $  ;to manage space for labels
   IMAGE_LOCATION=xr[0:1,0],$
   IMAGE_DIMENSIONS=[xr[0,1]-xr[0,0],xr[1,1]-xr[1,0]],$
   MAP_PROJECTION='Mercator', DIMENSIONS=[700,700])
```

Here we see the map, and the image. But, the map annotations
are clearly wrong. The latitude 40.1 which should be the center
of Fort Collins, Colorado is now located somewhere far north,

up near Yellowstone National Park in Wyoming!

The Coyote Graphics routines to display the same image:

```
map = Obj_New('cgMap', 'mercator', ELLIPSOID='wgs 84', /OnImage, $
  XRange=xrange, YRange=yrange)
cgDisplay, 700, 700
cgImage, googleimage, Margin=0.1
map -> Draw
cgMap_Grid, Map=map, /Box_Axes, Color='goldenrod'
cgMap_Continents, /USA, Map=map, Color='goldenrod'
```

Here are the two images, for those of you who don't want to
run the code:

   http://www.idlcoyote.com/misc/fg_google_map.png
   http://www.idlcoyote.com/misc/cg_google_map.png

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.")