
Subject: shifted .kml created with IDL

Posted by [titan](#) on Mon, 30 Jul 2012 12:25:01 GMT

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

Hi all,

I'm creating a .kml starting from a geotiff image. After reading this geotiff file I transformed it as a .png file and then I create the .kml file.

the code I'm using is the following

```
img_tiff=READ_TIFF(file_tiff, GEOTIFF=geotag)
png_fname='test' + tag + '.png'
WRITE_PNG, png_fname, img_tiff,/ORDER

name='kml_test'

openw, lun, kml_file, /get_lun
printf, lun, '<?xml version="1.0" encoding="UTF-8"?>'
printf, lun, '<kml xmlns="http://www.opengis.net/kml/2.2">'
printf, lun, '<GroundOverlay>'
printf, lun, name, format='("<name>", a, "</name>")'
printf, lun, '<Icon>'
printf, lun, png_fname, format='("<href>", a, "</href>")' ;
printf, lun, '</Icon>'
printf, lun, '<LatLonBox>'
printf, lun, ul_lat, format='("<north>", f10.4, "</north>")'
printf, lun, ul_lon, format='("<west>", f10.4, "</west>")'
printf, lun, lr_lat, format='("<south>", f10.4, "</south>")'
printf, lun, lr_lon, format='("<east>", f10.4, "</east>")'
printf, lun, '<rotation>0.0</rotation>'
printf, lun, '</LatLonBox>'
printf, lun, '<drawOrder>10</drawOrder>'
printf, lun, '</GroundOverlay>'
printf, lun, '</kml>'
free_lun, lun
PRINT,'KML file created'
```

The UL and LR coordinates I'm using are calculated in a previous section of the code and they are the following:

```
ul_lat = 47.653348
ul_lon = 8.9734384
lr_lat = 45.292876
lr_lon = 13.830713
```

My problem is that when I open the .kml file it is shifted and is not correctly overlaying with the features likes lake or border lines.

Even if I inserted the coordinates taken from google earth:

```
ul_lat_GE=47.750684
ul_lon_GE=9.015974
;
lr_lat_GE=45.222022
lr_lon_GE=13.612157
```

I'm still not able to correctly overlay my image :(

Could someone tell me what I'm doing wrong?

thanks in advance

Subject: Re: shifted .kml created with IDL
Posted by [titan](#) on Tue, 31 Jul 2012 07:09:59 GMT
[View Forum Message](#) <> [Reply to Message](#)

On Monday, July 30, 2012 4:23:18 PM UTC+2, titan wrote:

> On Monday, July 30, 2012 3:13:24 PM UTC+2, David Fanning wrote:

>

>> David Fanning writes:

>

>>

>

>>

>

>>

>

>>> I haven't made KML files, but when I work with Google Maps,

>

>>

>

>>> they always come back in a Mercator projection. I presume

>

>>

>

>>> that is what they are using.

>

>>

>

>>

>

>>

>

>> To navigate the Google Static Maps that I get from Google,

```
>
>>
>
>> I use a Mercator projection with a WGS84 datum:
>
>>
>
>>
>
>>
>
>>
>
>> mapCoord = Obj_New('cgMap', 105, Datum=8)
>
>>
>
>>
>
>>
>
>> 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.")
>
>
>
> Thanks a lot David,
>
> Tomorrow I will try using it as structure in the MAP_PROJ_INVERSE function and I will let you
know as soon as possible
>
>
>
> cheers
```

Hi David,
unfortunately it does not work or probably I'm still missing something :(
Using the following code I get coordinates that are pointing to an area near Corsica!!

```
mapCoord_Google_earth = MAP_PROJ_INIT('Mercator', Datum=8)
lonlat_Google_Earth = MAP_PROJ_INVERSE([xLL,yLL,xUR,yUR],
MAP_STRUCTURE=mapCoord_Google_earth)
```

```
lon_range_Google_Earth=[lonlat_Google_Earth[0,0],lonlat_Google_Earth[0,1]]
lat_range_Google_Earth=[lonlat_Google_Earth[1,0],lonlat_Google_Earth[1,1]]
```

```
UL_lat_Google_Earth=lonlat_Google_Earth[1,1]
UL_lon_Google_Earth=lonlat_Google_Earth[0,0]
```

```
LR_lat_Google_Earth=lonlat_Google_Earth[1,0]
LR_lon_Google_Earth=lonlat_Google_Earth[0,1]
```

```
print,'UL_lat_Google_Earth',UL_lat_Google_Earth
print,'UL_lon_Google_Earth',UL_lon_Google_Earth
print,'LR_lat_Google_Earth',LR_lat_Google_Earth
print,'LR_lon_Google_Earth',LR_lon_Google_Earth
```

```
UL_lat_Google_Earth    47.512672
UL_lon_Google_Earth    4.4728665
LR_lat_Google_Earth    45.054882
LR_lon_Google_Earth    7.7499209
```

What I'm doing wrong?!?

thanks

Titan writes:

```
> unfortunately it does not work or probably I'm still missing something :(
> Using the following code I get coordinates that are pointing to an area near Corsica!!
>
> mapCoord_Google_earth = MAP_PROJ_INIT('Mercator', Datum=8)
> lonlat_Google_Earth = MAP_PROJ_INVERSE([xLL,yLL,xUR,yUR],
MAP_STRUCTURE=mapCoord_Google_earth)
>
> lon_range_Google_Earth=[lonlat_Google_Earth[0,0],lonlat_Google_Earth[0,1]]
> lat_range_Google_Earth=[lonlat_Google_Earth[1,0],lonlat_Google_Earth[1,1]]
>
> UL_lat_Google_Earth=lonlat_Google_Earth[1,1]
> UL_lon_Google_Earth=lonlat_Google_Earth[0,0]
>
> LR_lat_Google_Earth=lonlat_Google_Earth[1,0]
> LR_lon_Google_Earth=lonlat_Google_Earth[0,1]
>
> print,'UL_lat_Google_Earth',UL_lat_Google_Earth
> print,'UL_lon_Google_Earth',UL_lon_Google_Earth
> print,'LR_lat_Google_Earth',LR_lat_Google_Earth
> print,'LR_lon_Google_Earth',LR_lon_Google_Earth
>
> UL_lat_Google_Earth      47.512672
> UL_lon_Google_Earth      4.4728665
> LR_lat_Google_Earth      45.054882
> LR_lon_Google_Earth      7.7499209
>
> What I'm doing wrong?!?
```

Pretty much everything, as far as I can tell. ;-)

Let's see. I think the goal was to take an image that is in a GeoTiff file and in one map projection (which one?) and convert it to another map projection (that we presume is a Mercator map projection), so we can make a KML file that overlaps.

So, we need:

1. A map structure that represents the image in the GeoTiff file.
2. A map structure that represents the map projection we want to eventually get the image into (Google Earth's

map projection.

3. Some method to convert from one map projection to the other.

If we are lucky, we can get the map structure for (1) from cgGeoMap. That will save us about 10 hours of work pawing though the GeoTiff documentation!

```
mapCoord = cgGeoMap(geoTiffFileName)
geoTiffMapStruct = mapCoord -> GetMapStruct()
```

We get the second map structure for (2) the way you set it up.

```
geMapStruct = Map_Proj_Init('Mercator', /GCTP, Datum=8)
```

For (3), we use Map_Proj_Image. We convert the XY projected meter values of the geoTiff image into the XY projected meter values of the Goggle Earth image. Then, convert those XY projected meter values into the lat/lon values we want to save in our KML file.

If we have done all this correctly (we make MANY assumptions that we hope are correct!), we will have shifted the lat/lon values in the GeoTiff image just a little bit, so that they align with the lat/lon values on a Google Earth map.

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: shifted .kml created with IDL
Posted by [titan](#) on Tue, 31 Jul 2012 14:51:32 GMT
[View Forum Message](#) <> [Reply to Message](#)

On Tuesday, July 31, 2012 1:56:40 PM UTC+2, David Fanning wrote:

> Titan writes:

>

>

>

>> unfortunately it does not work or probably I'm still missing something :(

```

>
>> Using the following code I get coordinates that are pointing to an area near Corsica!!
>
>>
>
>> mapCoord_Google_earth = MAP_PROJ_INIT('Mercator', Datum=8)
>
>> lonlat_Google_Earth = MAP_PROJ_INVERSE([xLL,yLL,xUR,yUR],
MAP_STRUCTURE=mapCoord_Google_earth)
>
>>
>
>> lon_range_Google_Earth=[lonlat_Google_Earth[0,0],lonlat_Google_Earth[0,1]]
>
>> lat_range_Google_Earth=[lonlat_Google_Earth[1,0],lonlat_Google_Earth[1,1]]
>
>>
>
>> UL_lat_Google_Earth=lonlat_Google_Earth[1,1]
>
>> UL_lon_Google_Earth=lonlat_Google_Earth[0,0]
>
>>
>
>> LR_lat_Google_Earth=lonlat_Google_Earth[1,0]
>
>> LR_lon_Google_Earth=lonlat_Google_Earth[0,1]
>
>>
>
>> print 'UL_lat_Google_Earth',UL_lat_Google_Earth
>
>> print 'UL_lon_Google_Earth',UL_lon_Google_Earth
>
>> print 'LR_lat_Google_Earth',LR_lat_Google_Earth
>
>> print 'LR_lon_Google_Earth',LR_lon_Google_Earth
>
>>
>
>> UL_lat_Google_Earth      47.512672
>
>> UL_lon_Google_Earth      4.4728665
>
>> LR_lat_Google_Earth      45.054882
>
>> LR_lon_Google_Earth      7.7499209
>

```

>>
>
>> What I'm doing wrong?!?
>
>
>
> Pretty much everything, as far as I can tell. ;-)
>
>
>
> Let's see. I think the goal was to take an image that is in
>
> a GeoTiff file and in one map projection (which one?) and
>
> convert it to another map projection (that we presume is
>
> a Mercator map projection), so we can make a KML file that
>
> overlaps.
>
>
>
> So, we need:
>
>
>
> 1. A map structure that represents the image in the GeoTiff
>
> file.
>
>
>
> 2. A map structure that represents the map projection we
>
> want to eventually get the image into (Google Earth's
>
> map projection.
>
>
> 3. Some method to convert from one map projection to the
>
> other.
>
>
>
> If we are lucky, we can get the map structure for (1) from
>

> cgGeoMap. That will save us about 10 hours of work pawing
>
> though the GeoTiff documentation!
>
>
>
> mapCoord = cgGeoMap(geoTiffFileName)
>
> geoTiffMapStruct = mapCoord -> GetMapStruct()
>
>
>
> We get the second map structure for (2) the way you set it up.
>
>
>
> geMapStruct = Map_Proj_Init('Mercator', /GCTP, Datum=8)
>
>
>
> For (3), we use Map_Proj_Image. We convert the XY projected
>
> meter values of the geoTiff image into the XY projected meter
>
> values of the Goggle Earth image. Then, convert those XY projected
>
> meter values into the lat/lon values we want to save in our
>
> KML file.
>
>
>
> If we have done all this correctly (we make MANY assumptions
>
> that we hope are correct!), we will have shifted the lat/lon
>
> values in the GeoTiff image just a little bit, so that they
>
> align with the lat/lon values on a Google Earth map.
>
>
>
> 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.")

Hi david,
thanks for your answer.
my tiff file is in UTM WGS 84 32N projection and I'm not sure if I understood well your step 3.

Translating (I hope correctly!!)your suggestions I have

```
mapCoord = cgGeoMap(tiff_image)
```

```
geoTiffMapStruct = mapCoord -> GetMapStruct()  
geMapStruct = Map_Proj_Init('Mercator', /GCTP, Datum=8)  
image=Read_Tiff(path_fname+tiff_image, GEOTIFF=geotag)
```

```
ul_lat = 47.653348  
ul_lon = 8.9734384  
lr_lat = 45.292876  
lr_lon = 13.830713
```

```
warp_r = Map_Proj_image(Reform(image[0,*,*]),[lr_lat,ul_lon,ul_lat,lr_lon],  
IMAGE_STRUCTURE=geoTiffMapStruct,MAP_STRUCTURE=geMapStruct,  
UVRANGE=warp_r_uvrage)  
s = Size(warp_r, /Dimensions)  
warp_g = Map_Proj_image(Reform(image[1,*,*]),[lr_lat,ul_lon,ul_lat,lr_lon],  
IMAGE_STRUCTURE=geoTiffMapStruct,MAP_STRUCTURE=geMapStruct,  
UVRANGE=warp_g_uvrage)  
warp_b = Map_Proj_image(Reform(image[2,*,*]),[lr_lat,ul_lon,ul_lat,lr_lon],  
IMAGE_STRUCTURE=geoTiffMapStruct,MAP_STRUCTURE=geMapStruct,  
UVRANGE=warp_b_uvrage)  
warp24 = BytArr(3, s[0], s[1])  
warp24[0, *, *] = ROTATE(warp_r,7)  
warp24[1, *, *] = ROTATE(warp_g,7)  
warp24[2, *, *] = ROTATE(warp_b,7)
```

in the third step should I convert again the warp24 file or not in the google earth projection?

thanks a lot for your patience

titan writes:

```
> Translating (I hope correctly!!)your suggestions I have
>
> mapCoord = cgGeoMap(tiff_image)
>
> geoTiffMapStruct = mapCoord -> GetMapStruct()
> geMapStruct = Map_Proj_Init('Mercator', /GCTP, Datum=8)
> image=Read_Tiff(path_fname+tiff_image, GEOTIFF=geotag)
>
> ul_lat =    47.653348
> ul_lon =    8.9734384
> lr_lat =    45.292876
> lr_lon =    13.830713
```

I'm not sure if these are the *centers* of the corner pixels or the outside edge. I would probably get this information directly from the GeoTiff file. If you did, you would be getting the outside edge boundaries.

In other words, you can get this information from the GeoTiff structure using tie points, the scale in the X and Y direction, and the size of the image. All info that is available for the GeoTiff file:

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

```
> warp_r = Map_Proj_image(Reform(image[0,*,*]),[lr_lat,ul_lon,ul_lat,lr_lon],
IMAGE_STRUCTURE=geoTiffMapStruct,MAP_STRUCTURE=geMapStruct,
UVRANGE=warp_r_uvrage)
> s = Size(warp_r, /Dimensions)
> warp_g = Map_Proj_image(Reform(image[1,*,*]),[lr_lat,ul_lon,ul_lat,lr_lon],
IMAGE_STRUCTURE=geoTiffMapStruct,MAP_STRUCTURE=geMapStruct,
UVRANGE=warp_g_uvrage)
> warp_b = Map_Proj_image(Reform(image[2,*,*]),[lr_lat,ul_lon,ul_lat,lr_lon],
IMAGE_STRUCTURE=geoTiffMapStruct,MAP_STRUCTURE=geMapStruct,
UVRANGE=warp_b_uvrage)
> warp24 = BytArr(3, s[0], s[1])
> warp24[0, *, *] = ROTATE(warp_r,7)
> warp24[1, *, *] = ROTATE(warp_g,7)
> warp24[2, *, *] = ROTATE(warp_b,7)
```

Unfortunately, the "limit" parameter in Map_Proj_Image is documented incorrectly in some on-line help versions. (I think this was fixed in

IDL 7.1.2 or something like that.) Those limits should be in terms of XY coordinates, not lat/lon coordinates!

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

(Latitude and longitude are not rectangular coordinates, so using those as the "limits" makes no sense at all, except in a few map projections.) These limits are the limits of your input image. The UVRANGE keyword returns the limits of your re-projected image. It is *these* limits that you want to turn into lat/lon values for your input to your KML file.

> in the third step should I convert again the warp24 file or not in the google earth projection?

You Inverse project the UVRANGE limits to lat/lon to pass this to your KML code.

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: shifted .kml created with IDL
Posted by [David Fanning](#) on Tue, 31 Jul 2012 16:36:37 GMT
[View Forum Message](#) <> [Reply to Message](#)

David Fanning writes:

> I'm not sure if these are the *centers* of the corner
> pixels or the outside edge. I would probably get this
> information directly from the GeoTiff file. If you did,
> you would be getting the outside edge boundaries.
>
> In other words, you can get this information from the
> GeoTiff structure using tie points, the scale in the
> X and Y direction, and the size of the image. All info
> that is available for the GeoTiff file:
>
> http://www.idlcoyote.com/map_tips/pixel_to_ll.html

All this boundary information is more easily obtained with the Coyote Library routine FindMapBoundary, of course:

```
void = FindMapBoundary(geoTiffFile, b, XRange=xr, YRange=yr)
```

The boundary is in the variable b. Or you can use the ranges. Same numbers, just presented differently, depending on what you need.

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: shifted .kml created with IDL

Posted by [titan](#) on Thu, 02 Aug 2012 07:00:48 GMT

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

On Tuesday, July 31, 2012 6:36:37 PM UTC+2, David Fanning wrote:

> David Fanning writes:

>

>

>

>> I'm not sure if these are the *centers* of the corner

>

>> pixels or the outside edge. I would probably get this

>

>> information directly from the GeoTiff file. If you did,

>

>> you would be getting the outside edge boundaries.

>

>>

>

>> In other words, you can get this information from the

>

>> GeoTiff structure using tie points, the scale in the

>

>> X and Y direction, and the size of the image. All info

>

>> that is available for the GeoTiff file:

>
>>
>
>> http://www.idlcoyote.com/map_tips/pixel_to_ll.html
>
>
>
> All this boundary information is more easily obtained
>
> with the Coyote Library routine FindMapBoundary, of course:
>
>
>
> void = FindMapBoundary(geoTiffFile, b, X RANGE=xr, Y RANGE=yr)
>
>
>
> The boundary is in the variable b. Or you can use the ranges.
>
> Same numbers, just presented differently, depending on what
>
> you need.
>
>
>
> 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.")

Hi David,
I tried to collect all your useful suggestions to get the rid of this problem but I think I'm still missing something and I'm getting confused :((Do I need holiday???)

I had two approaches:

1. I'm calculating the center of the corner pixels and from this (using http://www.idlcoyote.com/map_tips/pixel_to_ll.html) I calculated the lat and lon for all pixel of the image in the Google Earth projection

```
tiff_image=path_fname+tiff_image
```

```
;mapCoord = cgGeoMap(tiff_image)
```

```
;geoTiffMapStruct = mapCoord -> GetMapStruct()
```

```
geMapStruct = Map_Proj_Init('Mercator', /GCTP, Datum=8)
```

```
image=Read_Tiff(tiff_image, GEOTIFF=geotag)
```

```
image = Reverse(image, 2) ; Necessary for TIFF files.
```

```
s=Size(image, /Dimensions)
```

```
xscale = geotag.ModelPixelScaleTag[0]
```

```
yscale = geotag.ModelPixelScaleTag[1]
```

```
tp = geotag.ModelTiePointTag[3:4]
```

```
xOrigin = tp[0]
```

```
yOrigin = tp[1] - (yscale * s[1])
```

```
xEnd = xOrigin + (xscale * s[0])
```

```
yEnd = tp[1]
```

```
xCenter = (xEnd - xOrigin) / 2.0 + xOrigin
```

```
yCenter = (yEnd - yOrigin) / 2.0 + yOrigin
```

```
uvec = Findgen(s[0]) * xscale + xOrigin
```

```
vvec = Findgen(s[1]) * yscale + yOrigin
```

```
uarray = Rebin(uvec, s[0], s[1])
```

```
varray = Rebin(Reform(vvec, 1, s[1]), s[0], s[1])
```

```
lonlat_ge= MAP_PROJ_INVERSE(uarray, varray, MAP_STRUCTURE=geMapStruct)
```

```
minLat_ge = Min(lonlat_ge[1,*], MAX=maxLat_ge)
```

```
minLon_ge = Min(lonlat_ge[0,*], MAX=maxLon_ge)
```

```
limit_ge = [minLat_ge, minLon_ge, maxLat_ge, maxLon_ge]
```

```
print, limit_ge
```

```
39.646844    6.1113937    42.130700    6.1182209
```

2. Then I tried the FindMapBoundary function to get the border of my tiff image and then using in a correct way the map_proj_image function. After reading the tiff file and creating a warped image using the Google Earth map structure I convert the uvrage in lat/lon to be passed to my kml.

```

void = FindMapBoundary(tiff_image, b, X RANGE= xr, Y RANGE= yr)
print, 'boundary ', b
print, 'X RANGE ', xr
print, 'Y RANGE ', yr

tiffCoord = cgGeoMap(tiff_image)

tiffCoord -> SetProperty, X RANGE= xr, Y RANGE= yr

MercatorCoord = Obj_New('cgMap', "MERCATOR", DATUM=8)

warped_r = Map_Proj_Image(REFORM(image[0,*,*]), b, $
    IMAGE_STRUCTURE=tiffCoord->GetMapStruct(),
MAP_STRUCTURE=MercatorCoord->GetMapStruct(), $
    UVRANGE=uvout)

warped_g = Map_Proj_Image(REFORM(image[1,*,*]), b, $
    IMAGE_STRUCTURE=tiffCoord->GetMapStruct(),
MAP_STRUCTURE=MercatorCoord->GetMapStruct(), $
    UVRANGE=uvout)

warped_b = Map_Proj_Image(REFORM(image[2,*,*]), b, $
    IMAGE_STRUCTURE=tiffCoord->GetMapStruct(),
MAP_STRUCTURE=MercatorCoord->GetMapStruct(), $
    UVRANGE=uvout)
print,'uvout'
print,uvout

s=SIZE(warped_r, /DIMENSIONS)
warped = BytArr(3, s[0], s[1])
warped[0, *, *] = ROTATE(warped_r,7)
warped[1, *, *] = ROTATE(warped_g,7)
warped[2, *, *] = ROTATE(warped_b,7)

WRITE_PNG, path_fname+'tiff_warped2mercator.png', warped

lonlat_kml= MAP_PROJ_INVERSE(uvout, MAP_STRUCTURE=geMapStruct)

boundary      680317.23      4878686.4      1045117.2      5152286.4
X RANGE      680317.23      1045117.2
Y RANGE      4878686.4      5152286.4

uvout
      1252403.4      5414120.5      1789982.0      5829831.8

lonlat_kml

```



```
11.250532    43.859578
16.079682    46.500001
```

The problem is that I expected coordinate near to these

```
ul_lat =    47.653348
ul_lon =     8.9734384
lr_lat =    45.292876
lr_lon =    13.830713
```

which are the corners of the area in which I'm interest.

Am I still doing everything wrong?? :(

Thanks for your support!!

Subject: Re: shifted .kml created with IDL

Posted by [David Fanning](#) on Thu, 02 Aug 2012 21:33:19 GMT

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

Titan writes:

```
> lonlat_kml
>    11.250532    43.859578
>    16.079682    46.500001
>
>
> The problem is that I expected coordinates near to these
> ul_lat =    47.653348
> ul_lon =     8.9734384
> lr_lat =    45.292876
> lr_lon =    13.830713
> which are the corners of the area in which I'm interest.
> Am I still doing everything wrong?? :(
```

I would say the art of programming is reconciling what we expect with what we get. :-)

I don't know how to answer you questions. I had my own set of assumptions from the get-go. The first is that you described the problem correctly and accurately. But, since I've never *seen* the problem, I can't judge if my assumption is correct. I based my assumption on my experience that when things are a "little bit off" it usually involves a mismatch in map projection parameters.

If things are as bad as you now say they are, then I'm at a complete loss. I suppose you have good

reasons for expecting something different?

Can you confirm the lat/lon values at the corners of your geoTiff image (with, for example, Map_Proj_Forward) are what you expect them to be?

They should not change very much, I don't think, in this coordinate transformation from UTM to Mercator.

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: shifted .kml created with IDL
Posted by [titan](#) on Fri, 03 Aug 2012 13:21:47 GMT
[View Forum Message](#) <> [Reply to Message](#)

On Thursday, August 2, 2012 11:33:19 PM UTC+2, David Fanning wrote:

> Titan writes:

>

>

>

>> lonlat_kml

>

>> 11.250532 43.859578

>

>> 16.079682 46.500001

>

>>

>

>>

>

>> The problem is that I expected coordinates near to these

>

>> ul_lat = 47.653348

>

>> ul_lon = 8.9734384

>

```
>> lr_lat =    45.292876
>
>> lr_lon =    13.830713
>
>> which are the corners of the area in which I'm interest.
>
>> Am I still doing everything wrong?? :(
>
>
>
> I would say the art of programming is reconciling what
>
> we expect with what we get. :-)
>
>
>
> I don't know how to answer you questions. I had my
>
> own set of assumptions from the get-go. The first
>
> is that you described the problem correctly and
>
> accurately. But, since I've never *seen* the problem,
>
> I can't judge if my assumption is correct. I based
>
> my assumption on my experience that when things are
>
> a "little bit off" it usually involves a mismatch
>
> in map projection parameters.
>
>
>
> If things are as bad as you now say they are, then
>
> I'm at a complete loss. I suppose you have good
>
> reasons for expecting something different?
>
>
>
> Can you confirm the lat/lon values at the corners
>
> of your geoTiff image (with, for example,
>
> Map_Proj_Forward) are what you expect them to be?
>
```

>
>
> They should not change very much, I don't think,
>
> in this coordinate transformation from UTM to
>
> Mercator.
>
>
>
> 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.")

Ok David,
at least I can say that the code should work fine.. now I can move on to understand where is the
trick in my geotiff image.

thanks a lot for your answers, they were all very useful
cheers
