Subject: Re: Associating GeoTiFF tags with basic Mercator projection parameters? Posted by David Fanning on Tue, 04 Jan 2011 07:06:48 GMT

View Forum Message <> Reply to Message

Barry Lesht writes:

```
> I've been through the GeoTiFF threads here and, as often advised, have
> tried to make sense of the GeoTiFF standard (http://remotesensing.org/
> geotiff/spec/geotiff6.html#6.3.3) without success. I'm hoping that
> someone with knowledge of using IDL to write GeoTiff files can give me
 some help with the following problem.
>
> I have a set of images that were created by transforming satellite
 data onto a basic (single standard parallel at the equator) Mercator
> projection. The projected images were created using IDL with a
 minimal set of parameters; I know the size of the image (pixels,
 lines), its geographic limits, the central meridian, and that it is
  isotropic. Here is a header dump of the mapped files (which are hdf).
>
  Sample:Projection Category = "IDL";
            Sample:Projection Name = "Mercator";
>
            Sample:Limit = 38.8744010925293, -92.41079711914062,
>
  50.60269927978516, -75.86920166015625;
            Sample:Projection ID = 9;
>
            Sample:Latitude Center = 0.;
>
            Sample:Longitude Center = -84.13999938964844;
>
            Sample:Rotation = 0.;
>
            Sample: Position = 0., 0., 1., 1.;
>
            Sample: Isotropic = 1;
>
            Sample: Scale = 0.;
>
            Sample:Central Azimuth = 0.;
>
>
  I do some manipulation of these files in my code and need to create
  output mapped images of an extracted region in the form of GeoTiFF
> files that can be read by ArcMap. The problem is that I seem to
> unable to find the correct set of GeoTiFF tags to accomplish this.
  For example:
>
>
  g_{tags} = { }
>
         ModelPixelScaleTag: [0.016153906d, 0.011453418d, 0d], $
>
         ModelTiepointTag: [0, 0, 0, -84.813, 46.580, 0.], $
>
         GTModeTypeGeoKey: 1,
                                          $ : Projected
>
         GTRasterTypeGeoKey: 1,
                                           $: Pixel represents
>
>
  area
         GeographicTypeGeoKey: 4326,
                                             $: WGS84
>
         GeogLinearUnitsGeoKey: 9001,
                                             $; meters
>
         GeogAngularUnitsGeoKey: 9102,
                                              $; angular degree
>
         ProjCoordTransGeoKey: 7,
                                           $; Mercator
```

```
    ProjNatOriginLongGeoKey: -84.13999d, $
    ProjNatOriginLatGeoKey: 0.0 $
    WRITE_TIFF, tif_name, new_comp_image, /FLOAT, GEOTIFF=g_tags
    Produces files that "lack spatial reference data" when read by
    ArcMap. I'd sure appreciate some guidance - maybe there is a better
```

I'm not a GeoTiff expert, but every GeoTiff file I've ever dealt with had tie points in Cartesian coordinates (projected meters). It looks to me like yours are in units of latitude/longitude. In any case, you say they are in meters (9001), so...

You can see how I created a GeoTIFF image (once!) in this article:

http://www.dfanning.com/map_tips/goesmap.html

Cheers,

> way to accomplish this?

David

--

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

Subject: Re: Associating GeoTiFF tags with basic Mercator projection parameters? Posted by BLesht on Tue, 04 Jan 2011 17:42:49 GMT

View Forum Message <> Reply to Message

Thanks, David. I tried that early on, but went to the angular measure (adding the GeogAngularUnitsGeoKey) for some reason (maybe I saw a different example somewhere). I'll give it another go. By the way, have nice recollections of IDL course you taught way back "in the day." Still telling "Coyote stories?" Barry

Subject: Re: Associating GeoTiFF tags with basic Mercator projection parameters? Posted by BLesht on Wed, 05 Jan 2011 03:38:09 GMT

View Forum Message <> Reply to Message

Just to close this thread with something that may be of use to anyone

who has a similar problem, David's suggestion of following the steps described in his article

http://www.dfanning.com/map_tips/goesmap.html

was right on.

Because I already had a projected image, I skipped the warping part and started with using MAP_PROJ_INIT to set up the projection. I used the GCTP version of the Mercator projection and specified the equator as the TRUE_SCALE_LATITUDE. Because my files were based on a subset of the original map, I used the subset geographic boundaries to define the LIMIT vector in the MAP_PROJ_INIT and in the MAP_PROJ_FORWARD procedure. The resulting GeoTiFF files have the correct map information.

```
MercMap = MAP PROJ INIT(105, /GCTP.
                                                     $: Mercator
        ELLIPSOID = 8,
                                        $: WGS84
        LIMIT = [Ir_lat, Ir_lon, ul_lat, ul_lon], $; Corners
should be enough
        CENTER LONGITUDE = center lon,
                                                   $
        TRUE SCALE LATITUDE = 0.0)
uv = MAP_PROJ_FORWARD([left_lon, top_lon, right_lon, bottom_lon], $
             [left_lat, top_lat, right_lat, bottom_lat], $
            MAP STRUCTURE=MercMap)
xscale = ABS(uv[0,0] - uv[0,2])/(s[0])
yscale = ABS(uv[1,1] - uv[1,3])/(s[1])
tp = [uv[0,0], uv[1,1]]
g_tags = { $}
       ModelPixelScaleTag: [xscale, yscale, 0], $
       ModelTiepointTag: [0, 0, 0, tp, 0], $
       GTModeTypeGeoKey: 1,
                                        $; Geographic
       GTRasterTypeGeoKey: 1,
                                       $: Pixel represents
area
                                          $: WGS84
       GeographicTypeGeoKey: 4326,
       GeogLinearUnitsGeoKey: 9001,
                                         $; meters
       GeogAngularUnitsGeoKey: 9102,
                                          $; angular degree
       ProiectedCSTypeGeoKey: 32767,
                                           $;
       ProjectionGeoKay: 32767,
       ProiCoordTransGeoKey: 7,
                                       $: Mercator
       ProjLinearUnitsGeoKey: 9001,
                                        $:
       ProjNatOriginLongGeoKey: center_lon $
       ProjNatOriginLatGeoKey: 0.0,
                                        $
                                        $
       ProjFalseNorthingGeoKey: 0,
       ProjFalseEastingGeoKey: 0,
```

```
ProjScaleAtNatOriginGeoKey: 1. $
}
```

Thanks again, David.

On Jan 4, 11:42 am, Barry Lesht <ble...@gmail.com> wrote:

- > Thanks, David. I tried that early on, but went to the angular measure
- > (adding the GeogAngularUnitsGeoKey) for some reason (maybe I saw a
- > different example somewhere). I'll give it another go. By the way,
- > have nice recollections of IDL course you taught way back "in the
- > day." Still telling "Coyote stories?" Barry