Georeferencing an Image

Mr. Bryan Bagnall
SPAWAR Systems Center, Pacific
Phone: 619-553-4061
Email: bryan.bagnall@navy.mil

Mr. Sparta Cheung
SPAWAR Systems Center, Pacific
Phone: 619-553-5927
Email: sparta.cheung@navy.mil

Overview of Talk

- adf

Using Geometry Templates

- When OSSIM opens an image, it always checks to see if there is an external geometry file
- This geometry file defines the information necessary for OSSIM to calculate the latitude / longitude of each pixel
- These files are simple text documents
- ossim\trunk\ossim\etc\templates
  - bilinear_projection_template.geom
  - utm_projection_template.geom
  - geographic_projection_template.geom

Process for Georeferencing an Image

- Copy a template
- Rename the template to have the same name as your image, but with a .geom extension and place it in the same directory as your image
  - i.e. If your image is called **image.png**, you would name the geometry file **image.geom**
- Fill in the projection information required by the template
  - At this point we could open the image in OSSIM and it would be georeferenced, but it would only be georeferenced in OSSIM applications
- Use ossim-icp.exe to convert the image to an image with the georeference information stored inside

Process for Georeferencing an Image

- ossim-icp.exe is used to convert images between different datatype supported by OSSIM
- In our case, we are using it to convert image.png to a geotiff called image.tiff
- With the examples images provided above we would use the following command for ossim-icp.exe to create a geoTiff
  - ossim-icp.exe tiff_tiled image.png image.tiff
- image.tiff would be our final georeferenced product

Bilinear Projection Template

With only the four corners of the image, you can use this template to georeference your image

This is the easiest template to implement

Test it with barcos.png

Bilinear projection template

```
//---------------------------------------------------------------------------
// $Id: bilinear_projection_template.geom 7662 2005-06-15 16:36:23Z dburken $
//
// Description:
// This is a keyword list with minimum set of key words to make an
// ossimBilinearProjection.
//
// NOTE:
// The dpt's are in image space(x, y)
//
// The gpt's are in decimal degrees (latitude, longitude, height, datum)
//
// gpt0 should correspond to dpt0.
//
// It is best to have at least the four corners of the image.
//---------------------------------------------------------------------------
dpt0:  ( 0.0, 0.0)
dpt1:  ( 2047.0, 0.0 )
dpt2:  ( 2047.0, 2047.0 )
dpt3:  ( 0.0, 2027.0 )
gpt0:  ( 37.0, -117.0, 0.0, WGE )
gpt1:  ( 37.0, -116.0, 0.0, WGE )
gpt2:  ( 36.0, -116.0, 0.0, WGE )
gpt3:  ( 36.0, -117.0, 0.0, WGE )
type:  ossimBilinearProjection
```

Other Projections

The other projections have more complex template files, but can still be used to georeference an image

Two more examples of templates follow

UTM Projection Template

```
//---------------------------------------------------------------------------
// $Id: utm_projection_template.geom 9465 2006-08-28 18:53:59Z dburken $
//
// Description:
// This is a keyword list with minimum set of key words to make an
// ossimUtmProjection.
//---------------------------------------------------------------------------
central_meridian:  -87.000000000000000
datum:  WGE
ellipse_code:  WE
ellipse_name:  WGS 84
false_easting_northing:  ( 500000.000000000000000, 0.000000000000000 )
false_easting_northing_units:  meters
hemisphere:  N
major_axis:  6378137.000000000000000
minor_axis:  6356752.314199999906123
origin_latitude:  0.000000000000000
pixel_scale_units:  meters
pixel_scale_xy:  ( 5.0, 5.0 )

// point(tie center of pixel) or area (tie upper left corner of pixel).
pixel_type:  point

tie_point_units:  meters
tie_point_xy:  ( 138425.0, 4738765.0 )
type:  ossimUtmProjection
zone:  16
```

Geographic Projection Template

```
//---------------------------------------------------------------------------
// $Id: geographic_projection_template.geom 20209 2011-11-04 15:21:37Z dburken $
//
// Description:
// This is a keyword list with minimum set of key words to make an
// ossimEquDistCylProjection (commonly called geographic).
//
// NOTE:
// If decimal_degrees_per_pixel_lat equals decimal_degrees_per_pixel_lon then
// your origin_latitude should be 0.0 or at the equator.  If not, the
// decimal_degrees_per_pixel_lon =
// decimal_degrees_per_pixel_lat * cos(fabs(origin_latitude))
//---------------------------------------------------------------------------
central_meridian:  0.0
datum:  WGE
ellipse_code:  WE
ellipse_name:  WGS 84
false_easting_northing:  ( 0.000000000000000, 0.000000000000000 )
false_easting_northing_units:  meters
origin_latitude:  0.0
pixel_scale_units:  degrees
pixel_scale_xy:  ( 0.066666667, 0.066666667 )
// point(tie center of pixel) or area (tie upper left corner of pixel).
pixel_type:  point
tie_point_units:  degrees
tie_point_xy:  ( -91.416731292574454, 42.792911858047304 )
type:  ossimEquDistCylProjection
```

ossim-icp.exe Documentation

Usage: ossim-icp.exe [options] <output_type> <input_file> <output_file>

Description:
ossim-icp.exe copies any supported input image format to any supported output image format format

Options:

--disable-elev Will disable the elevation

--disable-notify Takes an argument. Arguments are ALL, WARN, NOTICE, INFO, FATAL, DEBUG. If you want multiple disables then just do multiple --disable-notify on the command line. All argument are case insensitive. Default is all are enabled.

--disable-plugin Will disable the plugin loader

--filter-spec <fname> This is an external file spec that describes a chain for filtering the input image.

--ossim-logfile takes a logfile as an argument. All output messages are redirected to the specified log file. By default there is no log file and all messages are enabled.

ossim-icp.exe Documentation

--pixel-type <type> Valid values: area or point, this will determine if the tie point is upper left corner of the upper left pixel (area) or the center of the upper left corner (point), default=point. NOTE: This option will only affect the tiff writer.

--reader-prop <string> Adds a property to send to the reader. format is name=value

--use-mask [<fname>] Optionally specify name of mask file to use for masking the input image. If no filename given, then the default mask filename is used.

--writer-prop <string> Adds a property to send to the writer. format is name=value

-K specify individual keywords to add to the preferences keyword list: name=value

-L or --end-line <n> Which end line do you wish to copy from the input. If none is given then max line is used

ossim-icp.exe Documentation

-P specify a preference file to load

-T specify the classes to trace, ex:
ossimInit|ossimImage.*
will trace ossimInit and all ossimImage
classes

-V or --version Display version information.

-a or --use-scalar-remapper Uses scalar remapper, transforms to 8-bit

-b or --bands <n,n...> uses the specified bands: ex. "1, 2, 4" will
select bands 1 2 and 4 of the input image.
Note: it is 1 based

-c or --compression-type <type> Uses compression. Currently valid for only
tiff output -c jpeg will use jpeg compression

-e or --entry <n> For multi image handlers which entry do you
wish to extract

-h or --help Display this information

-l or --start-line <n> Which start line do you wish to copy from the
input. If none is given then 0 is used

-o or --create-overview Creates and overview for the output image

ossim-icp.exe Documentation

-p or --end-sample <n> The end sample you wish to copy from the input. If none is given then max sample is used

-q or --compression-quality <n> Uses compression. Valid for jpeg type. default is 75 where 100 is best and 1 is worst

-r or --res-level <n> Which res level to extract from the input: ex -r 1 will get res level 1

-s or --start-sample <n> Which start sample do you wish to copy from the input. If none is given then 0 is used

-t or --create-thumbnail <n> Takes an argument which is the maximum pixel dimension desired.

-w or --tile-width <n> Defines the tile width for the handlers that support tiled output

ossim-icp.exe Documentation

Valid output writer types:

tiff_strip, tiff_strip_band_separate, tiff_tiled, tiff_tiled_band_separate, jpeg, general_raster_bip, general_raster_bil, general_raster_bsq, general_raster_bip_envi, general_raster_bil_envi, general_raster_bsq_envi, nitf_block_band_separate, nitf_block_band_sequential, nitf20_block_band_separate, nitf20_block_band_sequential, gdal_VRT, gdal_GTiff, gdal_NITF, gdal_HFA, gdal_ELAS, gdal_AAIGrid, gdal_DTED, gdal_PNG, gdal_JPEG, gdal_MEM, gdal_GIF, gdal_XPM, gdal_BMP, gdal_PCIDSK, gdal_PCRaster, gdal_ILWIS, gdal_SGI, gdal_SRTMHGT, gdal_Leveller, gdal_Terragen, gdal_HDF4Image, gdal_ERS, gdal_FIT, gdal_RMF, gdal_RST, gdal_INGR, gdal_GSAG, gdal_GSBG, gdal_R, gdal_PNM, gdal_ENVI, gdal_EHdr, gdal_PAux, gdal_MFF, gdal_MFF2, gdal_BT, gdal_IDA, gdal_USGSDEM, gdal_ADRG, gdal_BLX, dal_GeoRaster, , gdal_KMLSUPEROVERLAY, gdal_SAGA