

How to convert to GeoTIFF for Shikisai products

The Shikisai products can be converted to GeoTIFF, by the GDAL command and parameter according to image type. Please refer the following for example.

【Case1】 L1B、 L2 Scene Products

① Conversion of L2 SST (Sea surface Temperature) ---> Page.2

【Case2】 PNG image by HDFView

② Conversion of L2 SST (Sea surface Temperature) image by HDFView ---> Page.4

【Case3】 L3 Products

③ Conversion of L3 NDVI (Normalized Difference Vegetation Index) ---> Page.6

① Conversion of L2 SST (Sea surface Temperature) (1/2)

How to convert to GeoTIFF format from the Shikisai L2 scene products.

1) Get the dataset name by GDALINFO

`gdalinfo GC1SG1_202006080125H04910_L2SG_SSTDQ_1002.h5`

[Image file name.h5]

Driver: HDF5/Hierarchical Data Format Release 5

Files: GC1SG1_202006080125H04910_L2SG_SSTDQ_1002.h5

Size is 512, 512

.
.

Subdatasets:

SUBDATASET_1_NAME=HDF5:"GC1SG1_202006080125H04910_L2SG_SSTDQ_1002.h5"://Geometry_data/Latitude a.
SUBDATASET_1_DESC=[783x501] //Geometry_data/Latitude (32-bit floating-point)
SUBDATASET_2_NAME=HDF5:"GC1SG1_202006080125H04910_L2SG_SSTDQ_1002.h5"://Geometry_data/Longitude b.
SUBDATASET_2_DESC=[783x501] //Geometry_data/Longitude (32-bit floating-point)
SUBDATASET_3_NAME=HDF5:"GC1SG1_202006080125H04910_L2SG_SSTDQ_1002.h5"://Geometry_data/Obs_time
SUBDATASET_3_DESC=[783x501] //Geometry_data/Obs_time (16-bit integer)
SUBDATASET_4_NAME=HDF5:"GC1SG1_202006080125H04910_L2SG_SSTDQ_1002.h5"://Geometry_data/Sensor_zenith
SUBDATASET_4_DESC=[783x501] //Geometry_data/Sensor_zenith (16-bit integer)
SUBDATASET_5_NAME=HDF5:"GC1SG1_202006080125H04910_L2SG_SSTDQ_1002.h5"://Geometry_data/Solar_zenith
SUBDATASET_5_DESC=[783x501] //Geometry_data/Solar_zenith (16-bit integer)
SUBDATASET_6_NAME=HDF5:"GC1SG1_202006080125H04910_L2SG_SSTDQ_1002.h5"://Image_data/QA_flag
SUBDATASET_6_DESC=[7820x5000] //Image_data/QA_flag (16-bit unsigned integer)
SUBDATASET_7_NAME=HDF5:"GC1SG1_202006080125H04910_L2SG_SSTDQ_1002.h5"://Image_data/SST c.
SUBDATASET_7_DESC=[7820x5000] //Image_data/SST (16-bit unsigned integer)

2) Convert to VRT (GDAL Virtual Format) by GDAL_TRANSLATE

Create VRT files for latitude, longitude and SST data.

`gdal_translate -of VRT HDF5:"GC1SG1_202006080125H04910_L2SG_SSTDQ_1002.h5"://Geometry_data/Latitude out_latitude.vrt`

a.

[Arbitrary file name.vrt]

`gdal_translate -of VRT HDF5:"GC1SG1_202006080125H04910_L2SG_SSTDQ_1002.h5"://Geometry_data/Longitude out_longitude.vrt`

b.

[Arbitrary file name.vrt]

`gdal_translate -of VRT HDF5:"GC1SG1_202001050130A05010_L2SG_SSTDQ_2000.h5"://Image_data/SST out_SST.vrt`

c.

[Arbitrary file name.vrt]

① Conversion of L2 SST (Sea surface Temperature) (2/2)

3) Edit the VRT file

After add the geolocational data (Latitude and Longitude) to the SST VRT file, save by overwriting.

< before >

Insert here →
<VRTDataset rasterXSize="5000" rasterYSize="7820">
<Metadata>
<MDI key="Geometry_data_Grid_interval">2500 </MDI>
<MDI key="Geometry_data_Grid_interval_unit">meter</MDI>
:
:

< after >

<VRTDataset rasterXSize="5000" rasterYSize="7820">
<Metadata domain="GEOLOCATION">
 <MDI key="X_DATASET">**out_longitude.vrt**</MDI>
 <MDI key="X_BAND">1</MDI>
 <MDI key="Y_DATASET">**out_latitude.vrt**</MDI>
 <MDI key="Y_BAND">1</MDI>
 <MDI key="PIXEL_OFFSET">0.5</MDI>
 <MDI key="LINE_OFFSET">0.5</MDI>
 <MDI key="PIXEL_STEP">10</MDI>
 <MDI key="LINE_STEP">10</MDI>
</Metadata>
<Metadata>
<MDI key="Geometry_data_Grid_interval">2500 </MDI>
<MDI key="Geometry_data_Grid_interval_unit">meter</MDI>
:
:

**Bold: the file
name(edited by 2)**

4) GeoTIFF conversion and reprojection by GDALWARP

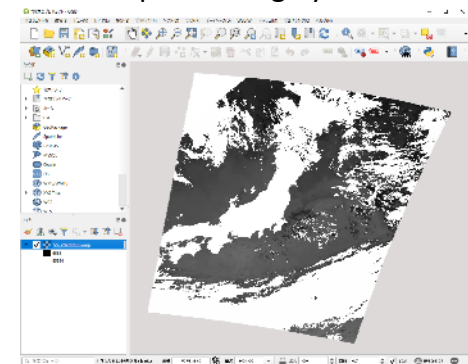
By the GDALWARP command, The SST VRT file convert to GeoTIFF format and reproject to EPSG:4326(Geodetic system: WGS84, Ellipsoidal model: WGS84, Projected coordinate system: latitude and longitude).

gdalwarp -of Gtiff -t_srs EPSG:4326 out_SST.vrt SST_output.tif

[the file name(edited by 3)).vrt]

[Arbitrary file name.tif]

< example viewing by QGIS >



② Conversion of L2 SST (Sea surface Temperature) image by HDFView (1/2)

How to convert to GeoTIFF format from the image data(PNG) output by HDFView.

1) Get the dataset name by GDALINFO

Get the latitude and longitude data set name of the original data for PNG conversion image.

gdalinfo GC1SG1_202006080125H04910_L2SG_SSTDQ_1002.h5

File name of original image

Driver: HDF5/Hierarchical Data Format Release 5

Files: GC1SG1_202006080125H04910_L2SG_SSTDQ_1002.h5

Size is 512, 512

.

.

Subdatasets:

SUBDATASET_1_NAME=HDF5:"GC1SG1_202006080125H04910_L2SG_SSTDQ_1002.h5"://Geometry_data/Latitude a.

SUBDATASET_1_DESC=[783x501] //Geometry_data/Latitude (32-bit floating-point)

SUBDATASET_2_NAME=HDF5:"GC1SG1_202006080125H04910_L2SG_SSTDQ_1002.h5"://Geometry_data/Longitude b.

SUBDATASET_2_DESC=[783x501] //Geometry_data/Longitude (32-bit floating-point)

.

.

SUBDATASET_7_NAME=HDF5:"GC1SG1_202006080125H04910_L2SG_SSTDQ_1002.h5"://Image_data/SST

SUBDATASET_7_DESC=[7820x5000] //Image_data/SST (16-bit unsigned integer)

2) Convert to VRT (GDAL Virtual Format) by GDAL_TRANSLATE

Create VRT files for latitude, longitude and SST image data(PNG) output by HDFView.

gdal_translate -of VRT HDF5:"GC1SG1_202006080125H04910_L2SG_SSTDQ_1002.h5"://Geometry_data/Latitude out_latitude.vrt

a.

gdal_translate -of VRT HDF5:"GC1SG1_202006080125H04910_L2SG_SSTDQ_1002.h5"://Geometry_data/Longitude out_longitude.vrt [Arbitrary file name.vrt]

b.

[Arbitrary file name.vrt]

gdal_translate -of SST.png out_SST.vrt

[Arbitrary file name.vrt]

PNG image of the same size
created by HDFView

② Conversion of L2 SST (Sea surface Temperature) image by HDFView (2/2)

3) Edit the VRT file

After add the geolocational data (Latitude and Longitude) to the SST VRT file, save by overwriting.

< before >

Insert here →
<VRTDataset rasterXSize="5000" rasterYSize="7820">
<VRTRasterBand dataType="Byte" band="1">
<ColorInterp>Red</ColorInterp>
<SimpleSource>
<SourceFilename relativeToVRT="1">SST.png</SourceFilename>
:
:

< after >

<VRTDataset rasterXSize="5000" rasterYSize="7820">
<Metadata domain="GEOLOCATION">
 <MDI key="X_DATASET">out_longitude.vrt</MDI>
 <MDI key="X_BAND">1</MDI>
 <MDI key="Y_DATASET">out_latitude.vrt</MDI>
 <MDI key="Y_BAND">1</MDI>
 <MDI key="PIXEL_OFFSET">0.5</MDI>
 <MDI key="LINE_OFFSET">0.5</MDI>
 <MDI key="PIXEL_STEP">10</MDI>
 <MDI key="LINE_STEP">10</MDI>
</Metadata>
<VRTRasterBand dataType="Byte" band="1">
<ColorInterp>Red</ColorInterp>
<SimpleSource>
<SourceFilename
 relativeToVRT="1">SST.png</SourceFilename>
:
:

Bold: the file
name(edited by 2)

4) GeoTIFF conversion and reprojection by GDALWARP

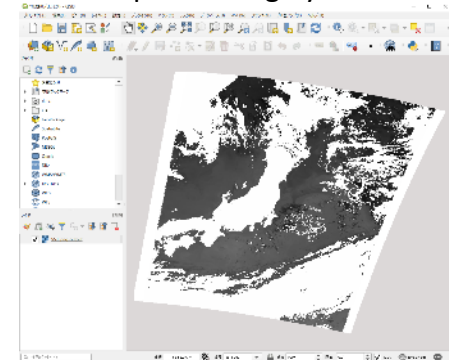
By the GDALWARP command, The SST VRT file convert to GeoTIFF format and reproject to EPSG:4326(Geodetic system: WGS84, Ellipsoidal model: WGS84, Projected coordinate system: latitude and longitude).

gdalwarp -of Gtiff -t_srs EPSG:4326 SST_png.vrt SST_png_output.tif

[the file name(edited by 3)).vrt]

[Arbitrary file name.tif]

< example viewing by QGIS >



③ Conversion of L3 NDVI (Normalized Difference Vegetation Index)

How to convert to GeoTIFF format from the Shikisai L3 products.

1) Get the dataset name by GDALINFO

`gdalinfo GC1SG1_20200401D01M_D0000_3MSG_NDVIF_1001.h5`

The image file name

Driver: HDF5/Hierarchical Data Format Release 5

Files: GC1SG1_20200401D01M_D0000_3MSG_NDVIF_1001.h5

Size is 512, 512

.

.

Subdatasets:

SUBDATASET_1_NAME=HDF5:"GC1SG1_20200401D01M_D0000_3MSG_NDVIF_1001.h5"://Image_data/NDVI_AVE

SUBDATASET_1_DESC=[4320x8640] //Image_data/NDVI_AVE (16-bit unsigned integer)

SUBDATASET_2_NAME=HDF5:"GC1SG1_20200401D01M_D0000_3MSG_NDVIF_1001.h5"://Image_data/NDVI_QA_flag

SUBDATASET_2_DESC=[4320x8640] //Image_data/NDVI_QA_flag (8-bit unsigned character)

2) Convert to GeoTIFF format by GDAL_TRANSLATE

By the following command, select the target dataset file, and convert to GeoTIFF format.

Projection code of input file

The range of input file (upper left(x,y), lower right(x,y))

`gdal_translate -of Gtiff -a_srs EPSG:4326 -a_ullr -180 90 180 -90`

`HDF5:"GC1SG1_20200401D01M_D0000_3MSG_NDVIF_1001.h5"://Image_data/NDVI_AVE NDVI_output.tif`

Input dataset file name

Output file name

< example viewing by QGIS >

