Transform GeoTIFF file to COG.

The folder contains three files: two Python scripts and one Jupyter Notebook.

1. tif_to_COG_translate.py transforms a GeoTIFF to a Cloud-Optimized-GeoTIFF (COG).
2. tif_to_COG_translate_notebook.ipynb is the Jupyter Notebook version of the script.
3. validate_cloud_optimized_geotiff.py is a Python script for check if a .tif file is or not a COG.

REQUIREMENTS:

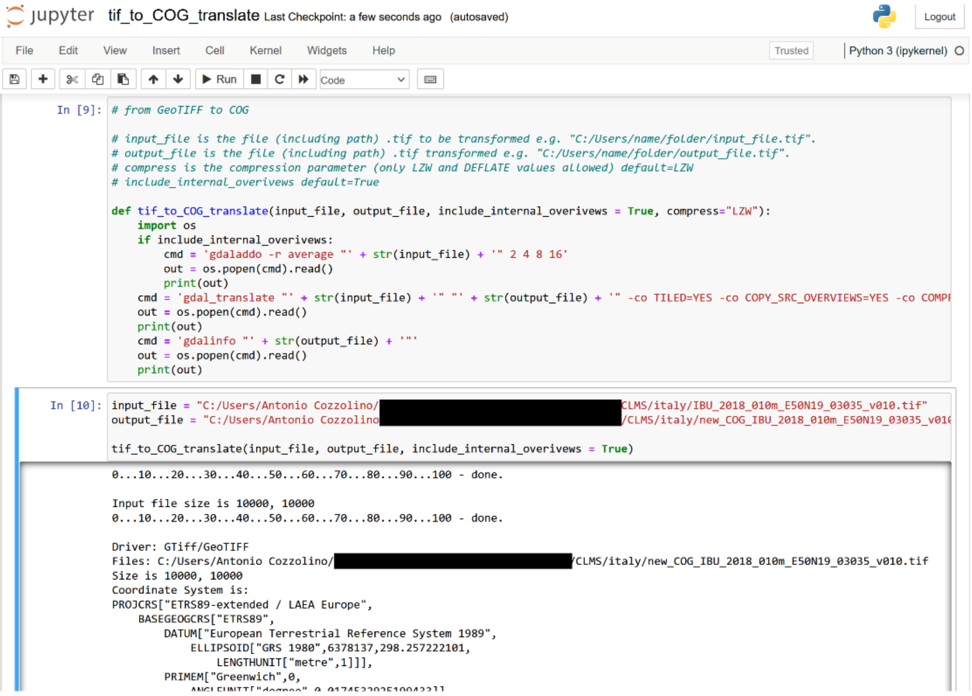
1. GDAL - https://gdal.org/index.html
2. PROJ - https://proj.org/about.html

The script tif_to_COG_translate.py transforms a GeoTIFF to COG using GDAL translator library and OS system call.

Parameters of function are:

1. Path of input file (e.g., *C:/User/……/input_file.tif*)
2. Path of output file (e.g., *C:/User/……/output_file.tif*)
3. Include internal overviews parameter.
4. Compression method. Typically, no compression, DEFLATE or LZW can be used for lossless, or JPEG for lossy. (Note that DEFLATE while more efficient than LZW can cause compatibility issues with some software packages)

tif_to_COG_translate.py can be use in this way:



```
In [9]: # from GeoTIFF to COG

        # input_file is the file (including path) .tif to be transformed e.g. "C:/Users/name/folder/input_file.tif".
        # output_file is the file (including path) .tif transformed e.g. "C:/Users/name/folder/output_file.tif".
        # compress is the compression parameter (only LZW and DEFLATE values allowed) default=LZW
        # include_internal_overivews default=True

        def tif_to_COG_translate(input_file, output_file, include_internal_overivews = True, compress="LZW"):
            import os
            if include_internal_overivews:
                cmd = 'gdaladdo -r average "' + str(input_file) + '" 2 4 8 16'
                out = os.popen(cmd).read()
                print(out)
            cmd = 'gdal_translate "' + str(input_file) + '" "' + str(output_file) + '" -co TILED=YES -co COPY_SRC_OVERVIEWS=YES -co COMPF
            out = os.popen(cmd).read()
            print(out)
            cmd = 'gdalinfo "' + str(output_file) + '"'
            out = os.popen(cmd).read()
            print(out)
```

```
In [10]: input_file = "C:/Users/Antonio Cozzolino/[redacted]CLMS/italy/IBU_2018_010m_E50N19_03035_v010.tif"
         output_file = "C:/Users/Antonio Cozzolino[redacted]/CLMS/italy/new_COG_IBU_2018_010m_E50N19_03035_v01(

         tif_to_COG_translate(input_file, output_file, include_internal_overivews = True)
```

```
0...10...20...30...40...50...60...70...80...90...100 - done.

Input file size is 10000, 10000
0...10...20...30...40...50...60...70...80...90...100 - done.

Driver: GTiff/GeoTIFF
Files: C:/Users/Antonio Cozzolino/[redacted]/CLMS/italy/new_COG_IBU_2018_010m_E50N19_03035_v010.tif
Size is 10000, 10000
Coordinate System is:
PROJCRS["ETRS89-extended / LAEA Europe",
    BASEGEOGCRS["ETRS89",
        DATUM["European Terrestrial Reference System 1989",
            ELLIPSOID["GRS 1980",6378137,298.257222101,
                LENGTHUNIT["metre",1]]],
        PRIMEM["Greenwich",0,
            ANGLEUNIT["degree",0.0174532925199433]]
```

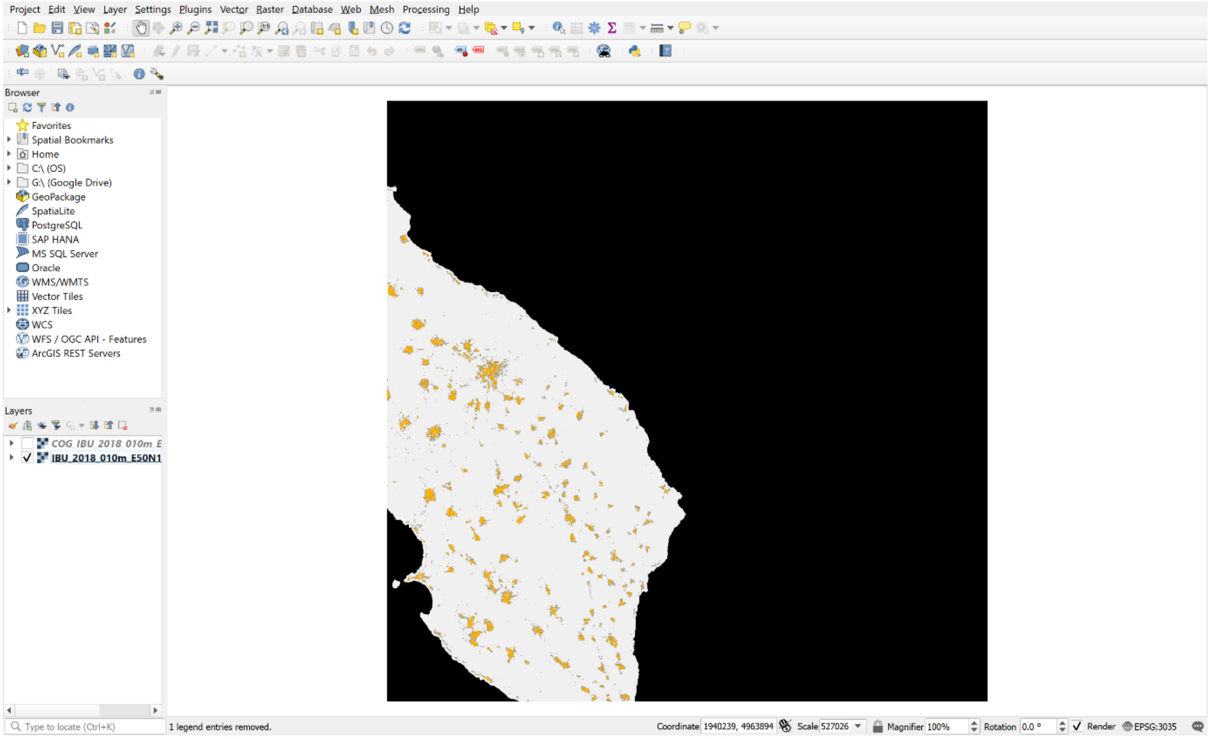
Figure 1 - Example of use

STEP BY STEP TRASFORMATION

This example was done using a GeoTIFF file downloaded from CLMS.

The steps performed are:

1. Viewing the initial .tif file in QGIS, in order to make a comparison with the respective COG obtained.
2. Checking whether the initial file is COG or not.
3. Transformation of the initial file into COG.
4. Displaying the COG .tif file in QGIS in order to compare it with the respective initial file.
5. Verify whether the resulting file is indeed COG.



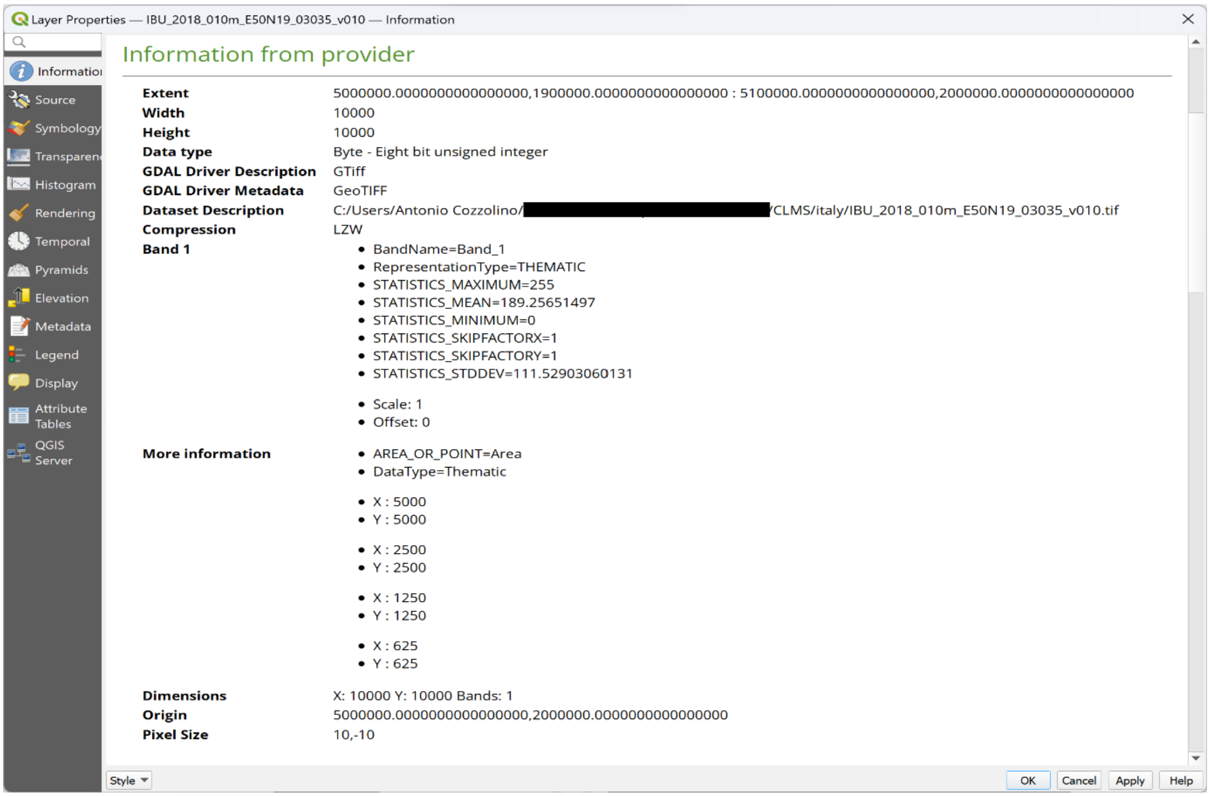
□ COG_IBU_2018_010m_E

☑ IBU_2018_010m_E50N1

1 legend entries removed.

Coordinate 1940239, 4963894 Scale 527026 Magnifier 100% Rotation 0.0 ° ✓ Render EPSG:3035

Figure 2 - NO-COG file in QGIS



Information from provider

Extent 5000000.0000000000000000,1900000.0000000000000000 : 5100000.0000000000000000,2000000.0000000000000000

Width 10000

Height 10000

Data type Byte - Eight bit unsigned integer

GDAL Driver Description GTiff

GDAL Driver Metadata GeoTIFF

Dataset Description C:/Users/Antonio Cozzolino/[redacted]/CLMS/italy/IBU_2018_010m_E50N19_03035_v010.tif

Compression LZW

Band 1

- BandName=Band_1
- RepresentationType=THEMATIC
- STATISTICS_MAXIMUM=255
- STATISTICS_MEAN=189.25651497
- STATISTICS_MINIMUM=0
- STATISTICS_SKIPFACTORX=1
- STATISTICS_SKIPFACTORY=1
- STATISTICS_STDDEV=111.52903060131

- Scale: 1
- Offset: 0

More information

- AREA_OR_POINT=Area
- DataType=Thematic

- X : 5000
- Y : 5000

- X : 2500
- Y : 2500

- X : 1250
- Y : 1250

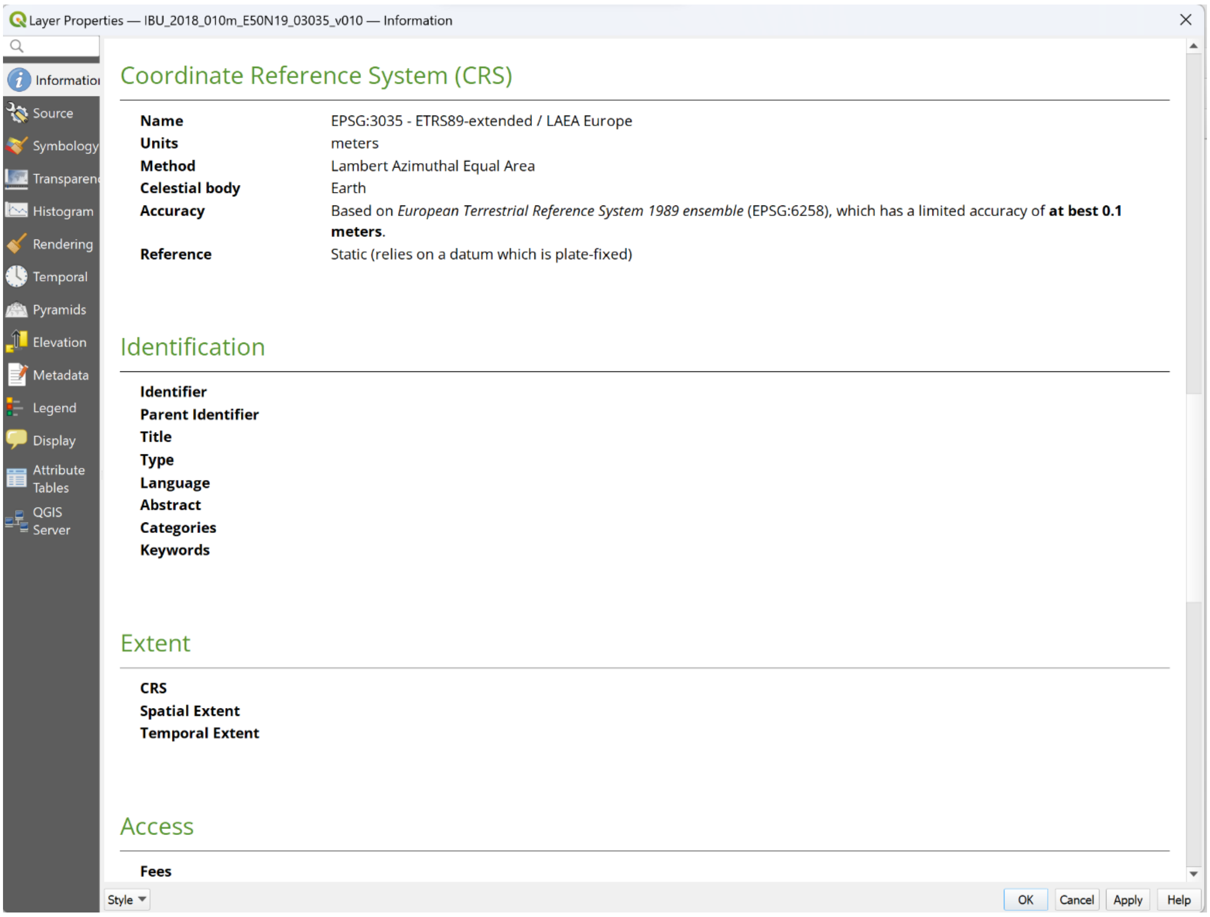
- X : 625
- Y : 625

Dimensions X: 10000 Y: 10000 Bands: 1

Origin 5000000.0000000000000000,2000000.0000000000000000

Pixel Size 10,-10

Figure 3 - NO-COG information



Coordinate Reference System (CRS)

Name EPSG:3035 - ETRS89-extended / LAEA Europe

Units meters

Method Lambert Azimuthal Equal Area

Celestial body Earth

Accuracy Based on *European Terrestrial Reference System 1989 ensemble* (EPSG:6258), which has a limited accuracy of **at best 0.1 meters**.

Reference Static (relies on a datum which is plate-fixed)

Identification

Identifier

Parent Identifier

Title

Type

Language

Abstract

Categories

Keywords

Extent

CRS

Spatial Extent

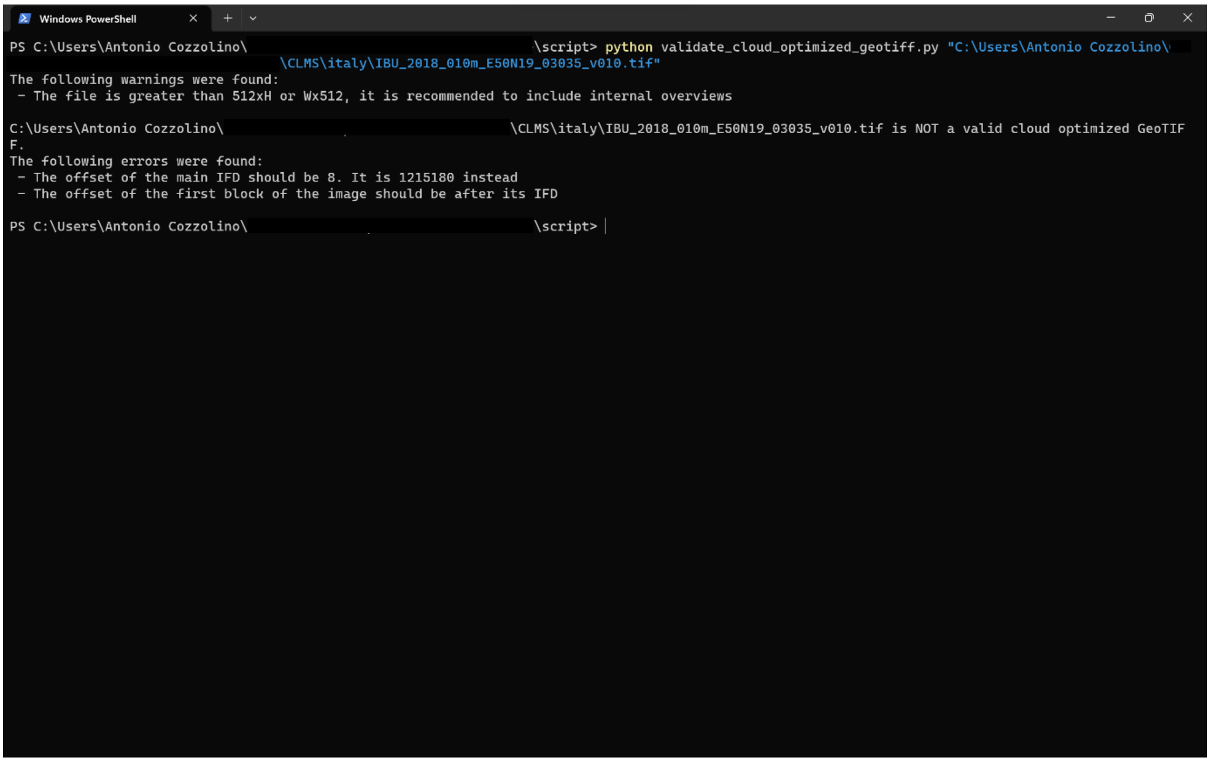
Temporal Extent

Access

Fees

Figure 4 - NO-COG CRS

Now, using validate_cloud_optimized_geotiff.py, we can see if the GeoTIFF file is really NO-COG.



```
PS C:\Users\Antonio Cozzolino\[redacted]\script> python validate_cloud_optimized_geotiff.py "C:\Users\Antonio Cozzolino\[redacted]\CLMS\italy\IBU_2018_010m_E50N19_03035_v010.tif"
The following warnings were found:
 - The file is greater than 512xH or Wx512, it is recommended to include internal overviews

C:\Users\Antonio Cozzolino\[redacted]\CLMS\italy\IBU_2018_010m_E50N19_03035_v010.tif is NOT a valid cloud optimized GeoTIF
F.
The following errors were found:
 - The offset of the main IFD should be 8. It is 1215180 instead
 - The offset of the first block of the image should be after its IFD

PS C:\Users\Antonio Cozzolino\[redacted]\script> |
```

The file is not COG, let us now proceed with the transformation, using the script tif_to_COG_translate.py.

Parameters used *are include_internal_overivews = True, compress="LZW".*

We obtained a new file which, when viewed in QGIS, appears:

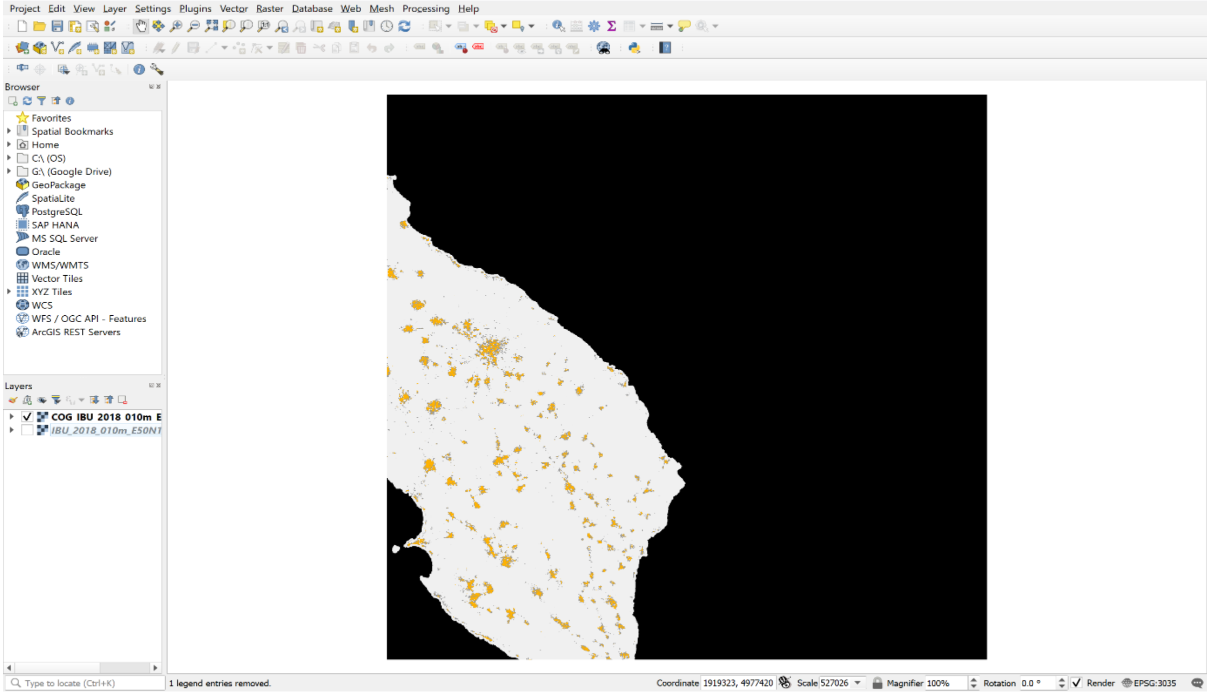
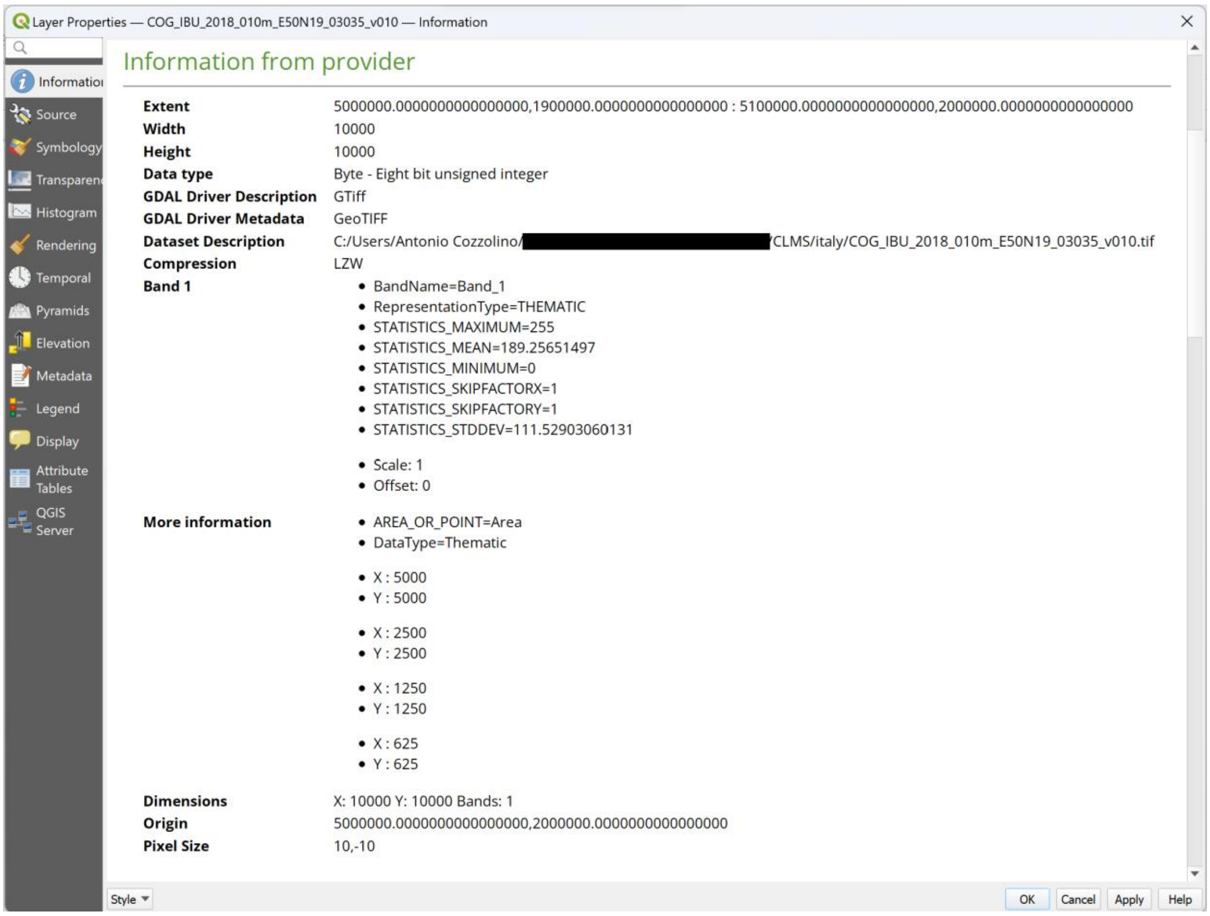


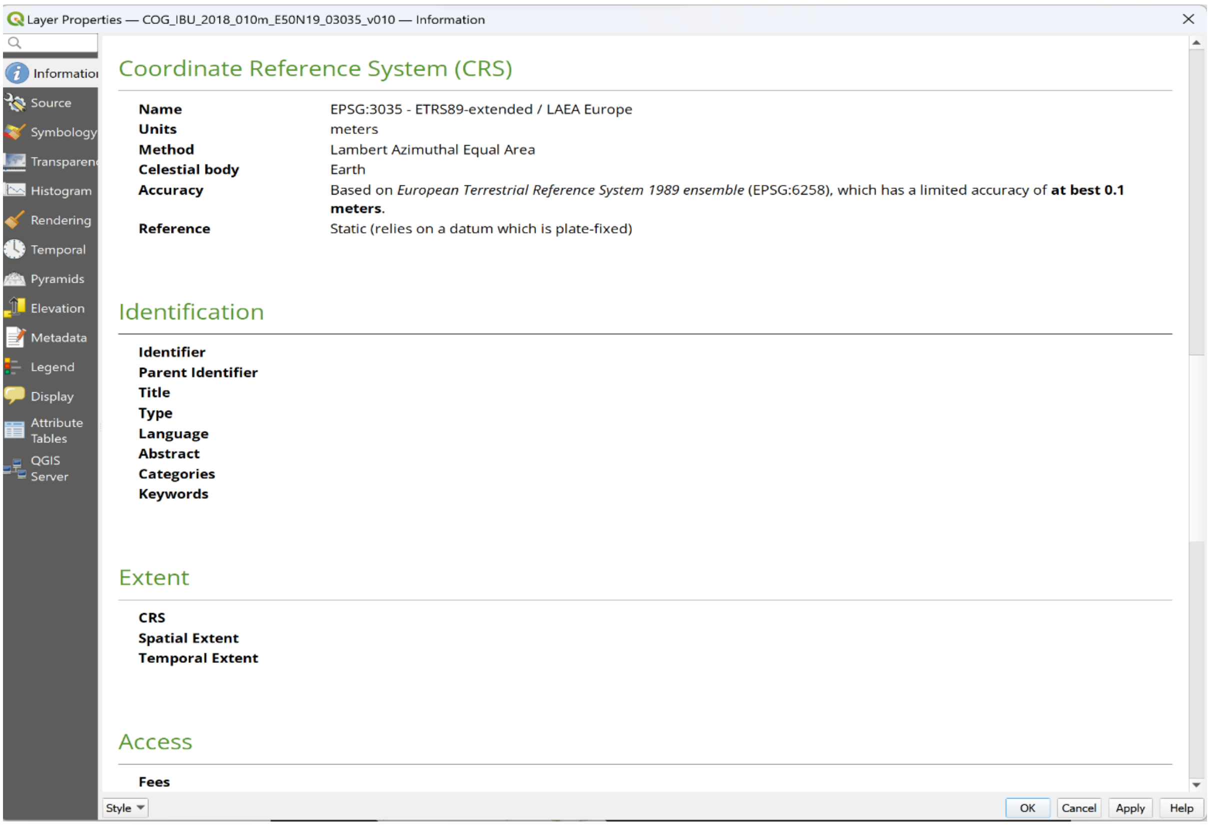
Figure 5 – COG file in QGIS



Information from provider

Extent	5000000.0000000000000000,1900000.0000000000000000 : 5100000.0000000000000000,2000000.0000000000000000
Width	10000
Height	10000
Data type	Byte - Eight bit unsigned integer
GDAL Driver Description	GTiff
GDAL Driver Metadata	GeoTIFF
Dataset Description	C:/Users/Antonio Cozzolino/[redacted]/CLMS/italy/COG_IBU_2018_010m_E50N19_03035_v010.tif
Compression	LZW
Band 1	• BandName=Band_1 • RepresentationType=THEMATIC • STATISTICS_MAXIMUM=255 • STATISTICS_MEAN=189.25651497 • STATISTICS_MINIMUM=0 • STATISTICS_SKIPFACTORX=1 • STATISTICS_SKIPFACTORY=1 • STATISTICS_STDDEV=111.52903060131 • Scale: 1 • Offset: 0
More information	• AREA_OR_POINT=Area • DataType=Thematic • X : 5000 • Y : 5000 • X : 2500 • Y : 2500 • X : 1250 • Y : 1250 • X : 625 • Y : 625
Dimensions	X: 10000 Y: 10000 Bands: 1
Origin	5000000.0000000000000000,2000000.0000000000000000
Pixel Size	10,-10

Figure 6 - COG file information



Coordinate Reference System (CRS)

Name	EPSG:3035 - ETRS89-extended / LAEA Europe
Units	meters
Method	Lambert Azimuthal Equal Area
Celestial body	Earth
Accuracy	Based on *European Terrestrial Reference System 1989 ensemble* (EPSG:6258), which has a limited accuracy of **at best 0.1 meters**.
Reference	Static (relies on a datum which is plate-fixed)

Identification

Identifier
Parent Identifier
Title
Type
Language
Abstract
Categories
Keywords

Extent

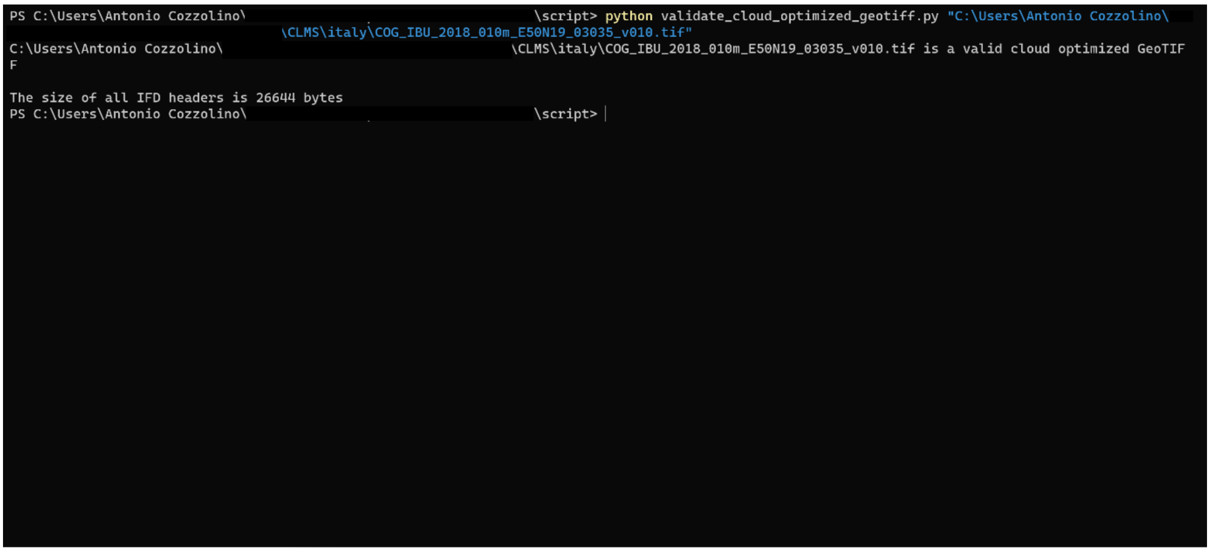
CRS
Spatial Extent
Temporal Extent

Access

Fees

Figure 7 - COG file CRS

Now, using the script using validate_cloud_optimized_geotiff.py we can check whether the resulting file is really COG.



```
PS C:\Users\Antonio Cozzolino\[redacted]\script> python validate_cloud_optimized_geotiff.py "C:\Users\Antonio Cozzolino\[redacted]
\CLMS\italy\COG_IBU_2018_010m_E50N19_03035_v010.tif"
C:\Users\Antonio Cozzolino\[redacted]\CLMS\italy\COG_IBU_2018_010m_E50N19_03035_v010.tif is a valid cloud optimized GeoTIF
F

The size of all IFD headers is 26644 bytes
PS C:\Users\Antonio Cozzolino\[redacted]\script>
```

The file we obtained is COG!