**Microwave & Hyperspectral RS & Digital Image Processing**

**Metadata:**

Metadata refers to the descriptive info. about the data itself. For example: we have a pic in tiff format i.e out data than the info. about this picture like its resolutions, format, no of bands, and other parameters of this image will called Metadata of this data.

**Reading TIFF Images:**

1. From PIL import Image
2. Img = Image.open(‘image.xyz’) #’image’ is file name and ‘xyz’ is the extension
3. from rasterio.plot import show
4. show(Img)

1) Python Imaging Library (PIL):

PIL for working with images having different formats & performing operation such as opening, manipulating, and saving images.

Image: Image is a module in PIL by which we can access to a wide range of functions and methods like “Image.open()” to open an image file.

1) From PIL import Image

2) Img = Image.open(‘image.xyz’)

In this line we created a variable ‘Img’ in which our opened image will be stored.

*Note: Image.open() is only used for single band TIFF file and normal images like JPEG.*

2) Rasterio:

It a python library used to working with geospatial raster data. It provides powerful set of tools for reading, writing, manipulating, and analyzing the raster data.

3) from rasterio.plot import show

4) show(Img)

The line 3 is importing a specific function called ‘show()’ from the module ‘rasterio.plot’ in the ‘rasterio’ library. ‘rasterio.plot’ contains functions related to plotting and visualizing of raster data. The function ‘show()’ is used to display the raster data.

**Another Method to Read TIFF file:**

1. import rasterio
2. img = rasterio.open(‘image.tif)
3. from rasterio.plot import show
4. show(img)

rasterio.open() is the function of rasterio module in rasterio library used to access raster data.

**Metadata About the TIFF File:**

1. import rasterio
2. img = rasterio.open(‘image.tif)
3. from rasterio.plot import show
4. width = img.width
5. height = img.height
6. num\_of\_bands = img.count
7. print(f'The resolution of the Image is {height} X {width}')
8. print(f'Number of bands in Image are {num\_of\_bands}')

The width, height, and count are the attribute that present in rasterio module used to check width, height and number of bands respectively.

Note: count attribute can only be used when open the raster image using ‘rasterio.open()’ and cannot be used on the data accessed by ‘image.open’.

We can also check the size of the image by using ‘size’ attribute in ‘Image’ module of ‘PIL’ library

as:

dimesions = img.size

print()

We can also display image by using ‘plt.show()’ function present in the ‘pyplot’ module of ‘matplotlib’ library as:

1. From PIL import Image
2. Img = Image.open(‘image.xyz’)
3. plt.imshow(Img)
4. plt.show()

*Note: ‘plt.imshow()’ is only applicable for the images opened by PIL and not valid for ‘rasterio’.*

**Calculate Image Statistics:**

To calculate image statistics like maximum and minimum pixel value, and std. deviation, we first have to convert our image in a numpy array.

1. import rasterio
2. img = rasterio.open(‘image.tif)
3. #Converting Image to a Numpy Array:
4. img\_array = img.read()
5. import numpy as np
6. max = np.max(img\_array)
7. min = np.min(img\_array)
8. mean = np.mean(img\_array)
9. std\_devi = np.std(img\_array)
10. print(f’The maximum pixel value is {max}’)
11. print(f’The minimum pixel value is {min}’)
12. print(f’The mean pixel value is {mean}’)
13. print(f’The Standard deviation of pixel values is { std\_devi })

In line 4 we used ‘img.read()’ function. This function lies in ‘rasterio’ module & is only valid for the images opened by ‘rasterio’. This function returns an array containing the pixel values of specific band. This function has different forms depending on how we want to read data, for example:

1. To read all the bands:

data = img.read () #no argument provided

1. To read a specific band:

data = img.read(1)

We could also use ‘numpy.array()’ which is a function of ‘numpy’ library used to create array. It returns a multi-dimensional array where each dimension corresponds to a different band and the spatial dimension of the raster data.

*Note: Numpy array is only valid for the image opened by PIL.*

In line 5, we impoted ‘numpy’ library as np which means we don’t have to write the word ‘numpy’ instead of which we can use just ‘np’. ‘numpy’ library provides many mathematical tools like std(), min(), and max() etc.

The max() in line 6 and min() in line 7, will return the maximum and minimum pixel value, and mean() in line 8 and std() in line 9 will provide mean pixel value and standard deviation between the pixels.

**Reducing the Size of the Image:**

1. #reading TIFF file by opencv
2. import cv2
3. img = cv2.imread('composite.tif')

OpenCV or CV2:

CV2 or OpenCV (open source computer vision library) provides a wide range of tools or functions for tasks related to image processing, machine learning, and object detection and tracking etc.

In Image processing it provides tools for image manipulation, filtering, transformation, and enhancement.

In the above 2 lines code, first we imported ‘cv2’ library to use its functions, than we use its function ‘imread()’ which is used to access images.

**Dimensions of the image:**

1. #reading TIFF file by opencv
2. import cv2
3. img = cv2.imread('composite.tif')
4. (width,height,Bands) = img.shape[:]

In line 4, ‘shape’ attribute is used which lies in ‘opencv’ library. It returns a tuple of three elements i.e height width and number of colors present in an image i.e number of bands.

Here ‘shape[:]’ means we are accessing all the three elements of the tuple and storing them in the variables width, height and Bands respectively.

**Resizing Image:**

1. #reading TIFF file by opencv
2. import cv2
3. img = cv2.imread('composite.tif')
4. (width,height,Bands) = img.shape[:]
5. Resize\_img = cv2.resize(img,(width/2,height/2), interpolation = cv2.INTER\_CUBIC)

‘resize()’ is the function in cv2 used to resize the dimensions of an image typically to make it larger or smaller. Its syntax is as follows:

cv2.resize(image,(new\_width,new\_height), interpolation = interpolation\_method)

where:

‘image’ is the variable in which our image is stored.

‘(new\_width,new\_height)’ is the tuple that specifies new dimensions.

*Note: The new dimensions should always be an integer.*

‘interpolation’ it is parameter that specifies the method used for estimating the pixel values in the resized image. It affects the quality the quality & smoothness of resized image.

Some common interpolation methods are:

* cv2.INTER\_NEAREST (fastest, least accurate)
* cv2.INTER\_LINEAR (better than NEAREST, but less smooth)
* cv2.INTER\_CUBIC (balance between quality and smoothness)
* cv2.INTER\_LANCZOS4 (slower, highest quality)