Plots Segmentation - User Manual

Version 1

Table of Contents

Detection Plots Version 1
User Manual produced by Susan L. Palacios Salcedo
Publication date: August 2021

OVERVIEW

Segmentation plots of multispectral aerial image is performed using two scripts developed in Python language.

Developed Scripts:

Main scripts for the execution of the algorithm.

- cluster_cip.py.
- plot_seg_cip

CHAPTER 1. CONFIGURATION PACKAGES IN ANACONDA

To use Python scripts, the user must install Anaconda and all the necessary packages. To edit the scprits, we recommended to install Visual Studio Code or Wordpad or Notepad.

Anaconda version required: Latest version

Required Libraries:

The following libraries list have been used in the plugin development.

- Python 3.9.6.
- Opencv 4.5.2
- numpy 1.21.1
- matplotlib 3.4.2
- gdal 3.3.1
- scikit-image 0.18.2
- rasterio 1.2.6
- pandas 1.3.1

Libraries Installation:

1. Install Anaconda (https://www.anaconda.com/products/individual-d)

2. In the bottom *Start Menu* look for: Anaconda Prompt (Anaconda3), right click and choose *Run as administrator* (Fig. 1). We should allow the system to run as administrator.

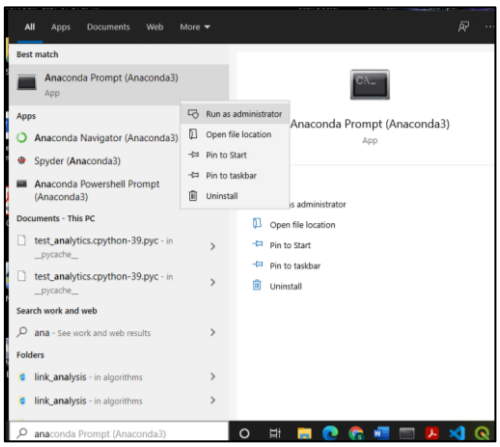

Fig. 1

3. A command line window will be opened (Fig. 2)

Fig. 2

4. Must be addressed to the work folder using the command *'cd'*. In this example the work folder is C:\Users\SUSAN (Fig. 3)

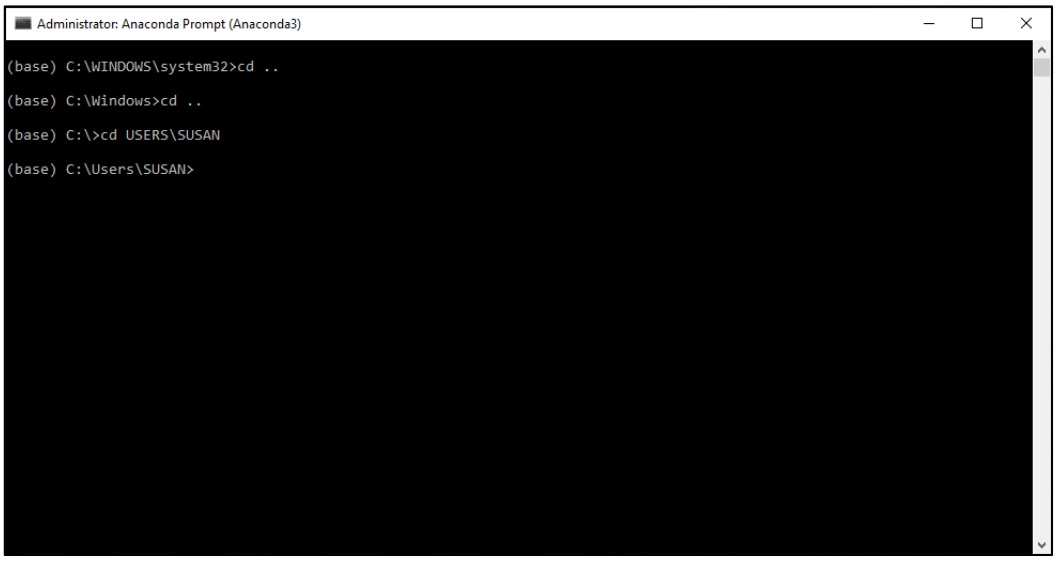

Fig. 3

5. Then, the git package must be installed to clone the folder where the codes are in the Git HUB. To do this, enter the following line in the command window (Fig. 4) and press enter.

```
conda install -c anaconda git
```

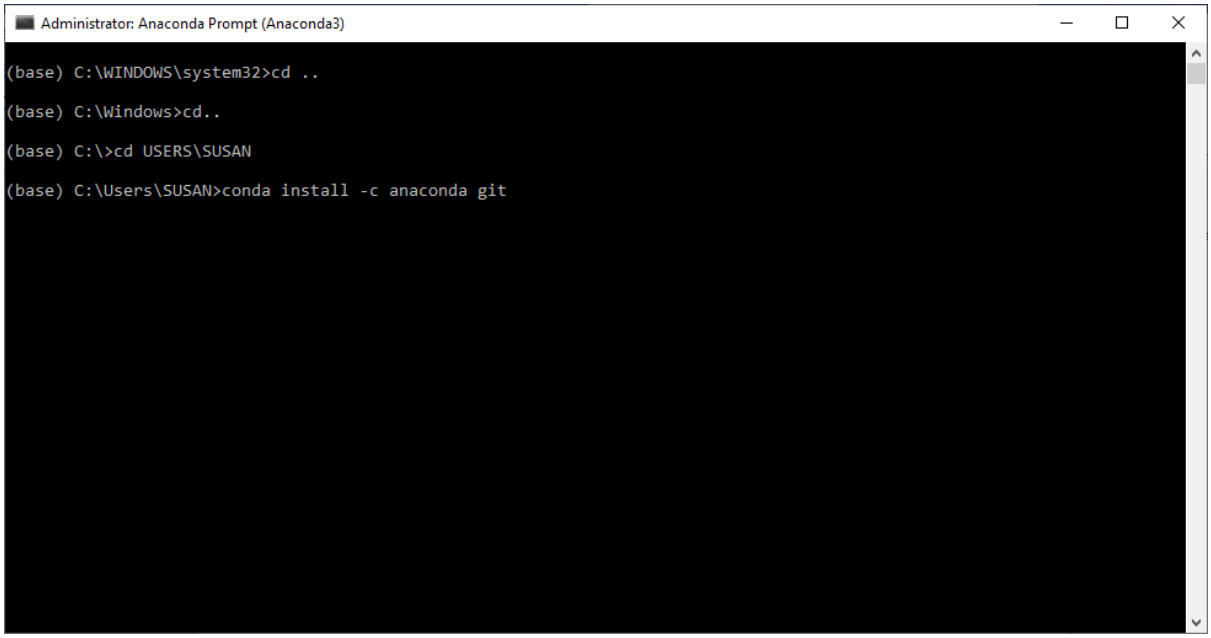

Fig. 4

6. When the git package has been installed. The folder must be cloned with the following command line (Fig. 5) and press enter:

   `git clone https://github.com/spalaciossalcedo/Plot-Segmentation.git`

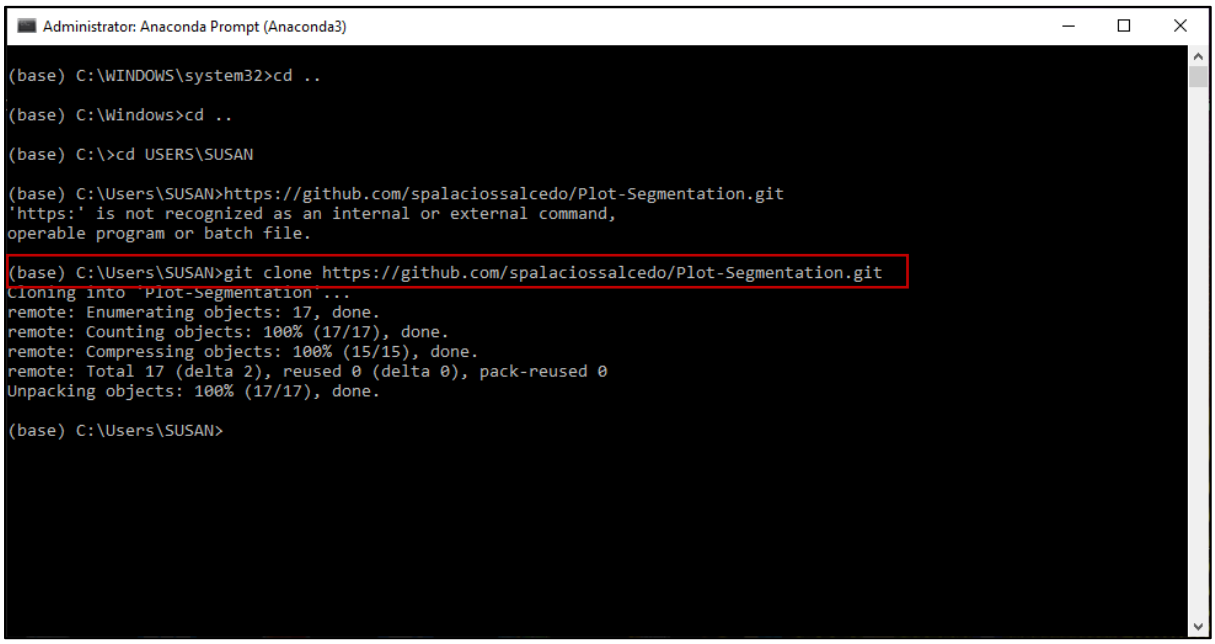

Fig. 5

7. Enter into the 'Plot-Segmentation' folder with the following command (Fig. 6)

   `cd Plot-Segmentation`

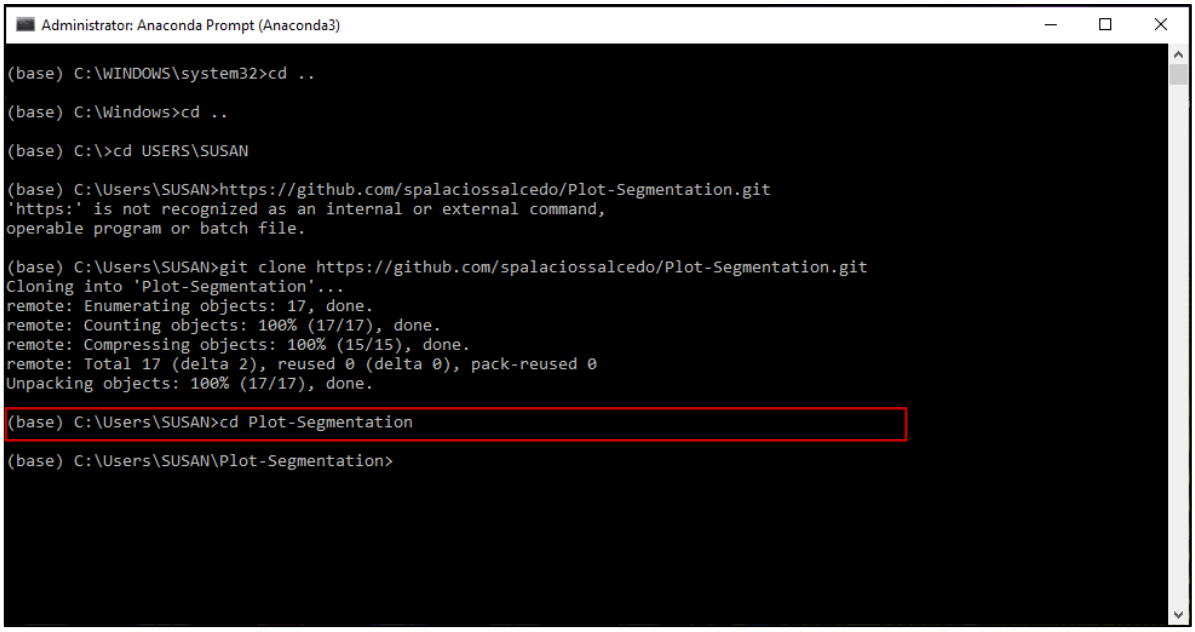

Fig. 6

8. To install the required libraries, use the following command line:

   ```
   conda env create -f cipseg.yml
   ```

   ```
   (base) C:\Users\SUSAN\Plot-Segmentation>conda env create -f cipseg.yml
   ```

9. After the installation is complete, the variable has to be activated using the following command. When the variable is activated on the command line it will appear as shown in (Fig. 7):

   ```
   conda activate cipseg
   ```

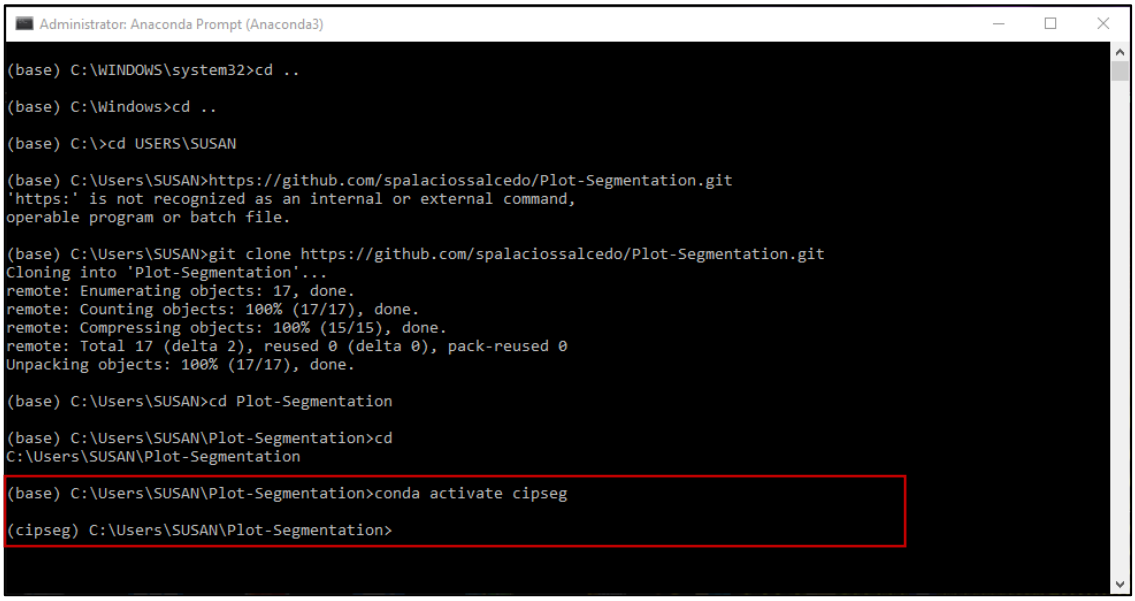

Fig. 7

CHAPTER 2. DATA FOLDER IN AND DATA FOLDER OUT CONFIGURATION

1. In the 'Plot-Segmentation' folder, a 'data' folder must be created where the image(s) to be processed will be saved. In addition, a 'data_output' folder must be created, where the results will be saved (Fig. 8).

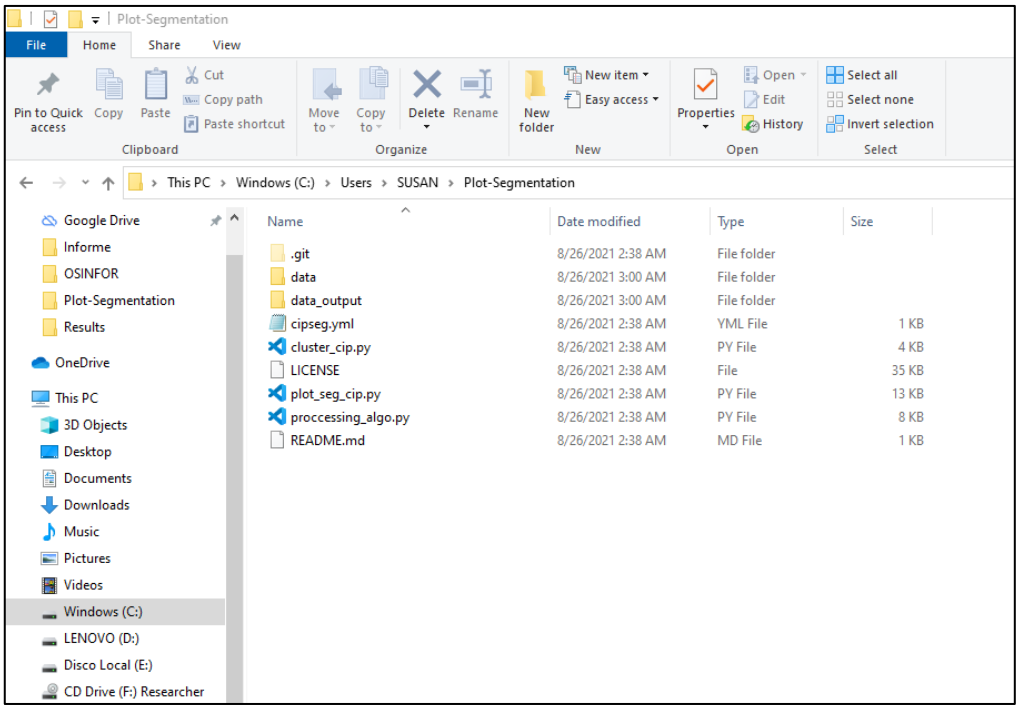

Fig. 8

CHAPTER 3. WORKFLOW

A workflow diagram is shown below and then the steps to run the scripts are described.

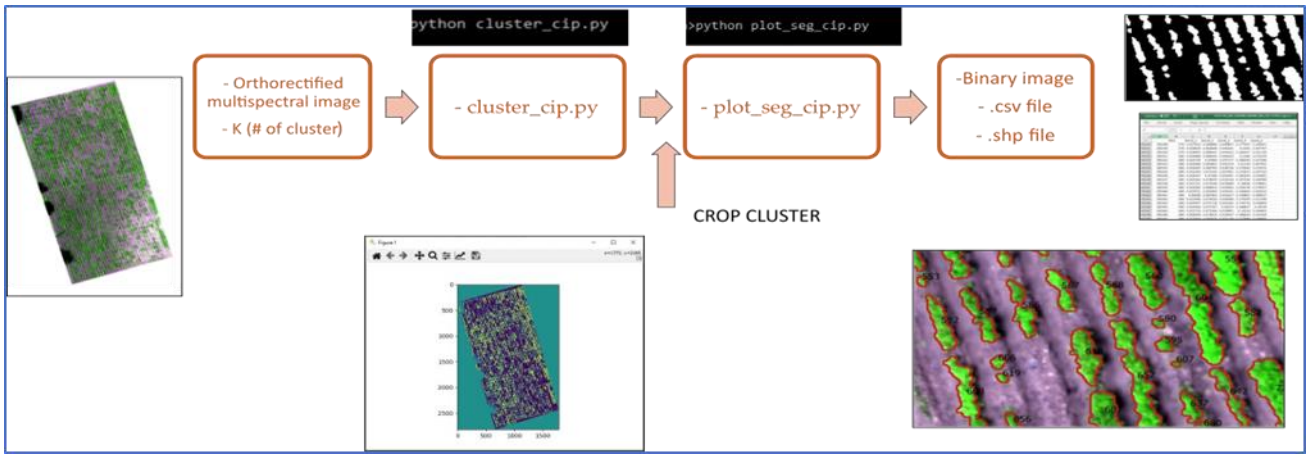

Fig. 9

1. In the ‘data’ folder we save the image to segment (Fig. 12), the preferred image should show only the crop area, it is recommended to cut the image using QGIS (Fig.11).

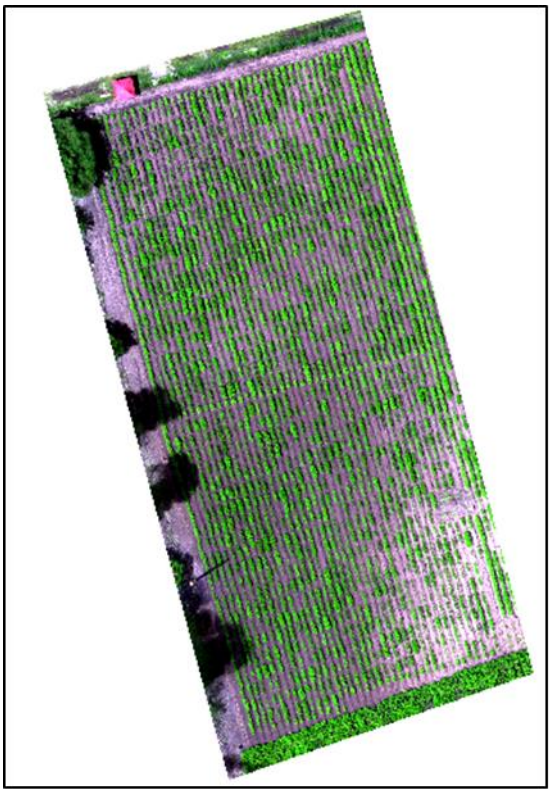

Fig. 10

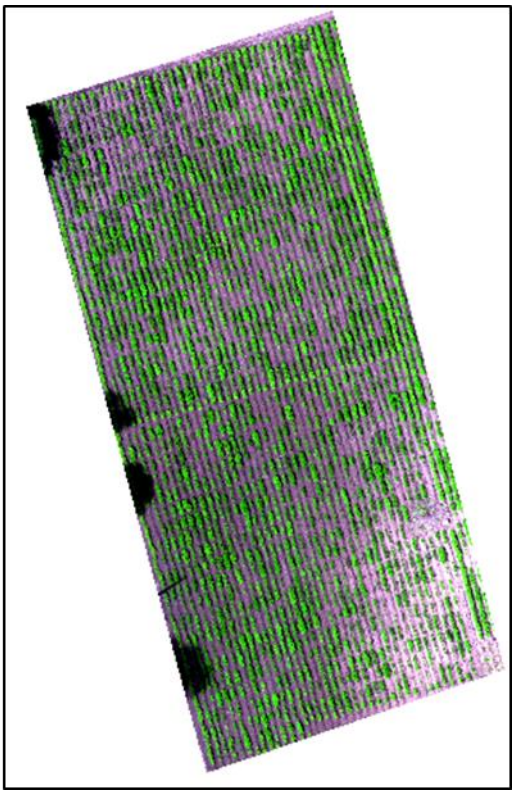

Fig. 11

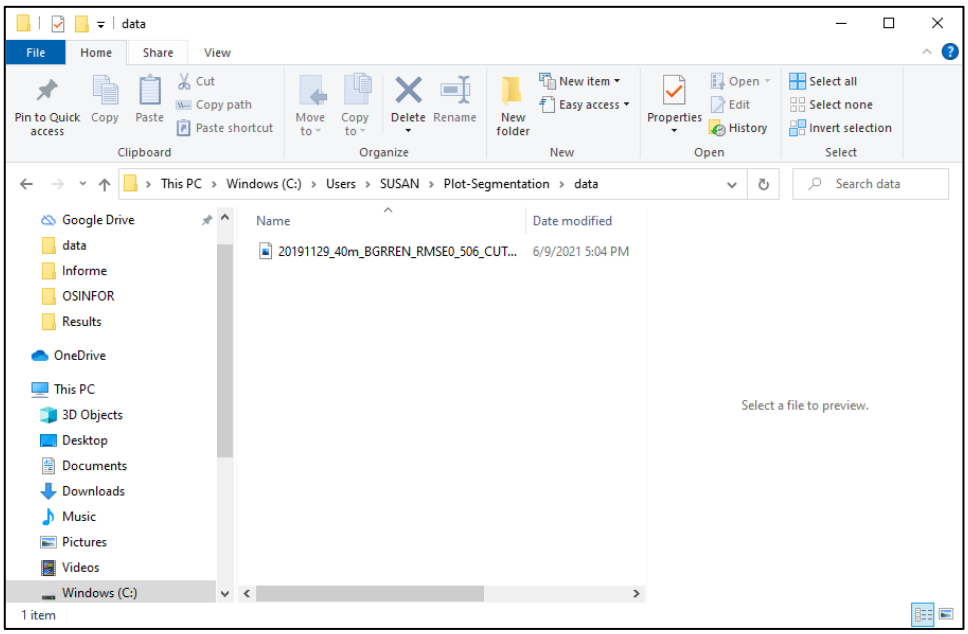

Fig. 12

2. In the working folder ‘Plot-Segmentation’, open the script ‘cluster_cip.py’ with WordPad, Notepad or Visual Studio Code. Then enter the location of the image to be processed that is in the data folder and the cluster number (k), it is recommended to use K = 3 and k = 4. Choose the value of (k) that allows to have the pixels of the plants in a single cluster (Fig. 13). After modifying, the file should be saved.

   - `inImg_path ='data/20191129_40m_BGRREN_RMSE0_506_CUT_CUTSP.tif'`
   - `k=3`

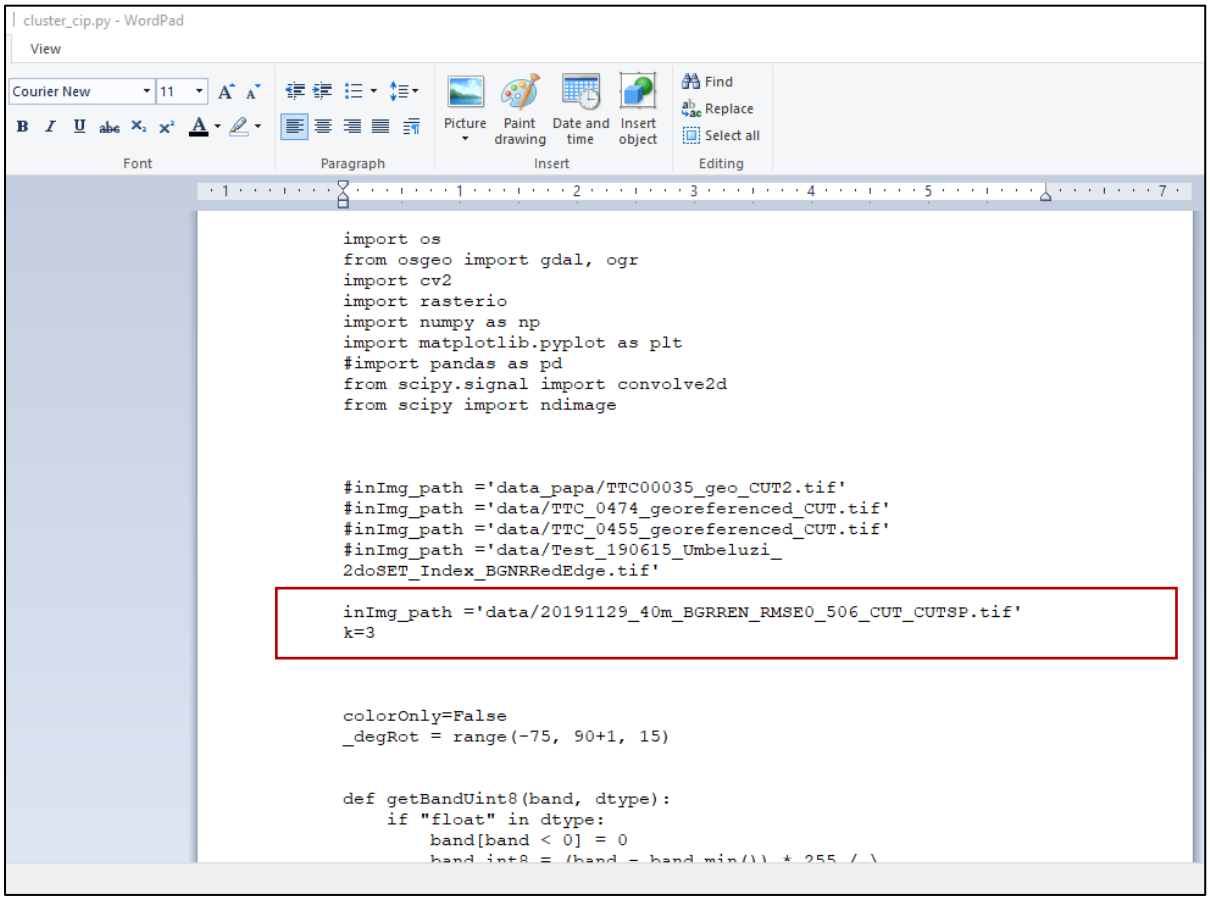

Fig. 13

3. Now the first script (cluster_cip.py) will be run to visualize the clusters. In the command line window, you must type the following and press enter.

```
python cluster_cip.py
```

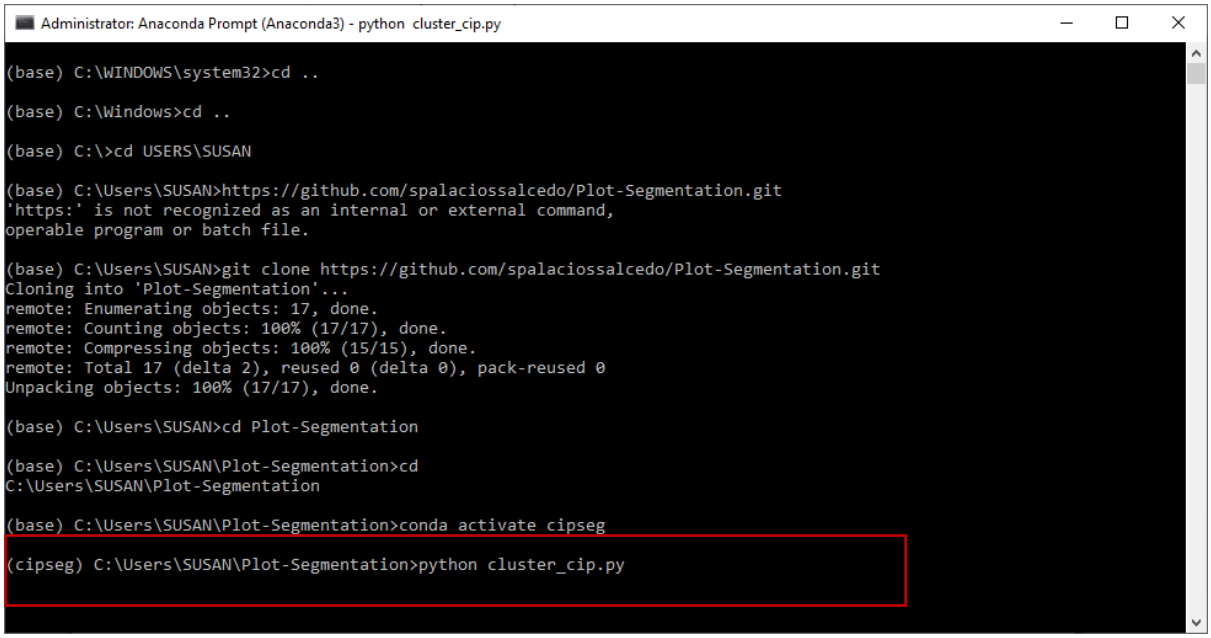

Fig. 14

4. When the script is executed, a window will appear, where it will be shown in the upper right corner to which cluster each pixel belongs (Fig. 15). The course be placed on the pixel and the cluster number will be indicated. In the example shown, it was identified that the cluster of plants is number 2. After identifying the cluster, the image must be closed. To run the following script.

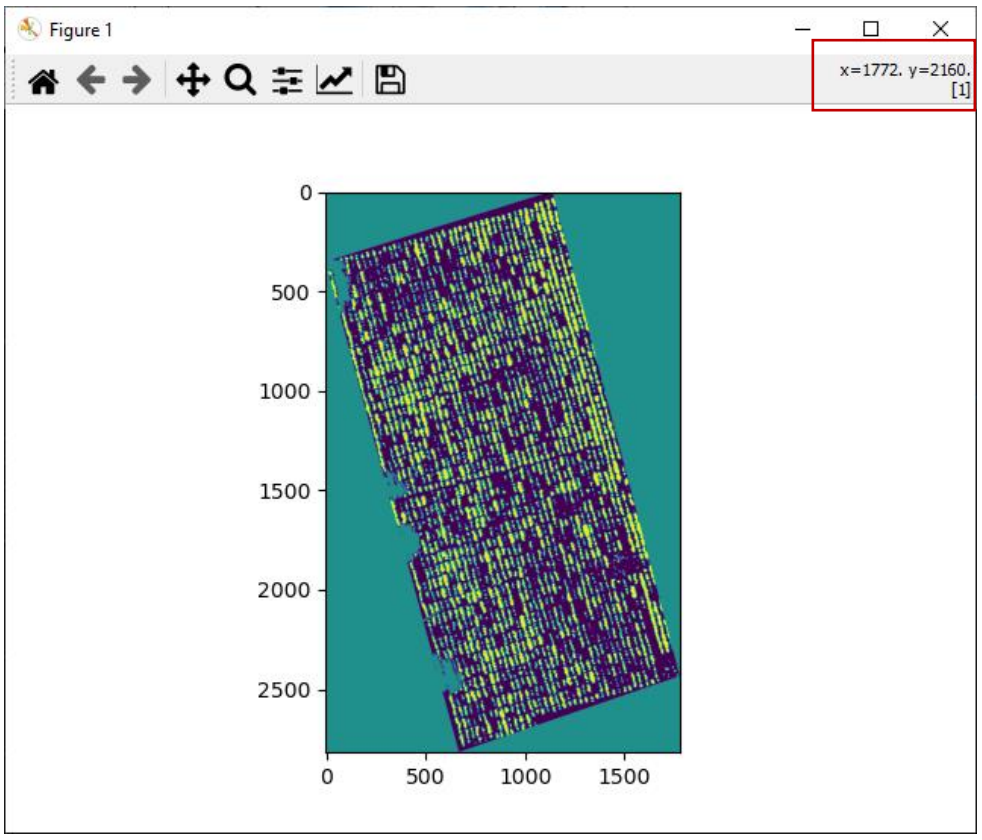

Fig. 15

5. Before executing the second script (plot_seg_cip.py), must be indicated in the script: the image address, the cluster number, the cluster that belongs to the plant, the output folder where a folder with the name of the plant will be created, image and the results files will be saved. After updating this data, it must be saved. In our example we will enter the following (Fig. 16):

```
-inImg_path ='data/20191129_40m_BGRREN_RMSE0_506_CUT_CUTSP.tif'
-output_folder='data_output'
-k=3
-crop_cluster=2
```

```
import os
from osgeo import gdal, ogr, osr
import cv2
import rasterio
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
#import pandas as pd
from scipy.signal import convolve2d
from scipy import ndimage
import proccessing_algo

#inImg_path ='data_papa/TTC00035_geo_CUT2.tif'
#inImg_path ='data/TTC_0474_georeferenced_CUT.tif'
#inImg_path ='data/TTC_0455_georeferenced_CUT.tif'
#inImg_path ='data/20191129_40m_BGRREN_RMSE0_506_CUT_CUTSP.tif'
#inImg_path ='data/Test_190615_Umbeluzi_
2doSET_Index_BGNRRedEdge.tif'

#features=[0,1,2]

#Input Image
inImg_path ='data/TTC_0474_georeferenced_CUT.tif'
#output folder
output_folder='data_output'
#Number of cluster
k=4
#Cluster of crop
crop_cluster=2
```

Fig. 16

6. Now the second script will be executed, to do this write the following in the command line window:

```
python plot_seg_cip.py
```

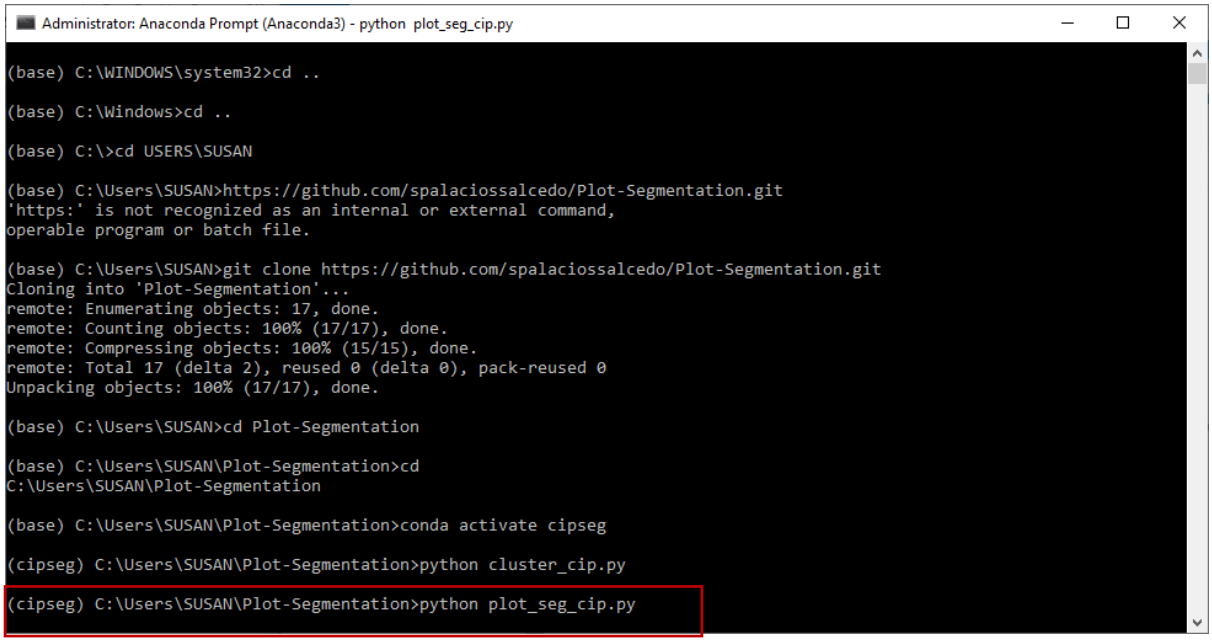

Fig. 17

7. Then in the folder 'data_output', shown a folder created with the name of the image (Fig. 18). The results of the algorithm are saved in this folder (Fig. 19).

- A .csv file where the information of each band related to the plot it belongs to is found.
- A .shp file where the segmented plots are saved.
- A binary image where pixels with a value of '1' indicate plant pixels, while pixels with a value of '0' indicate what is not plant.

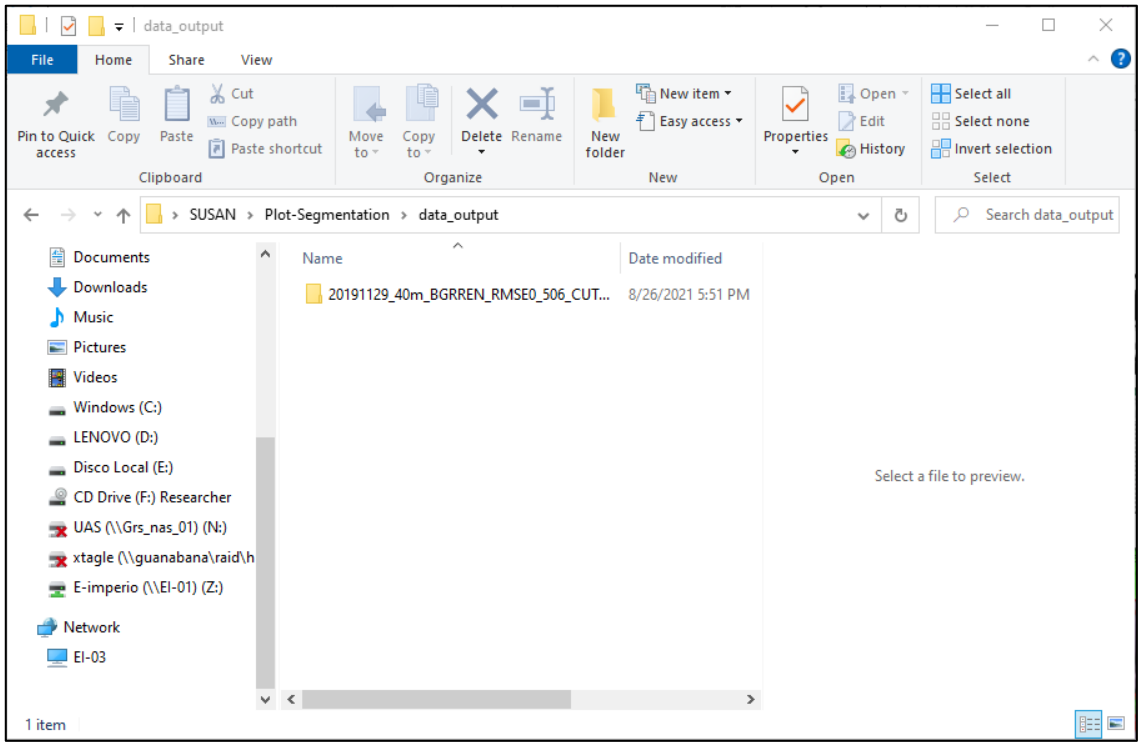

Fig. 18

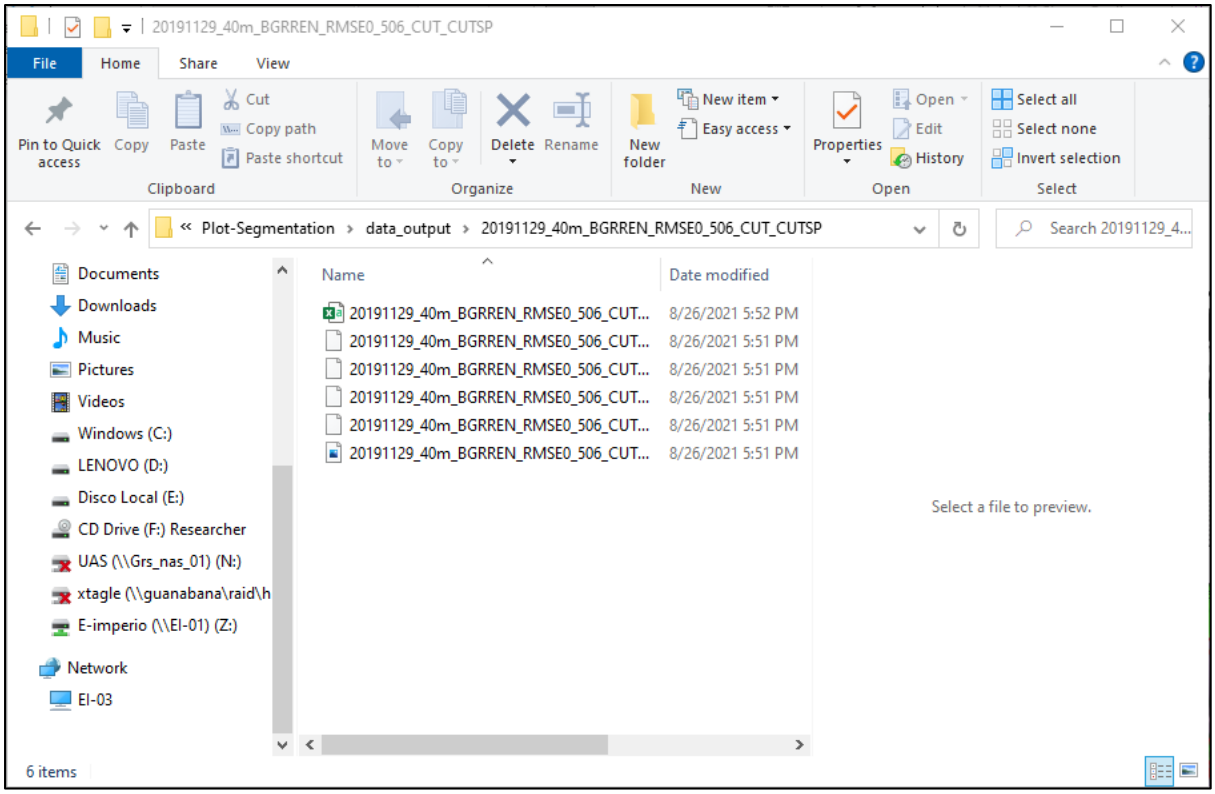

Fig. 19

8. The .shp file must be uploaded to QGIS to view the labels and verify the parcels.

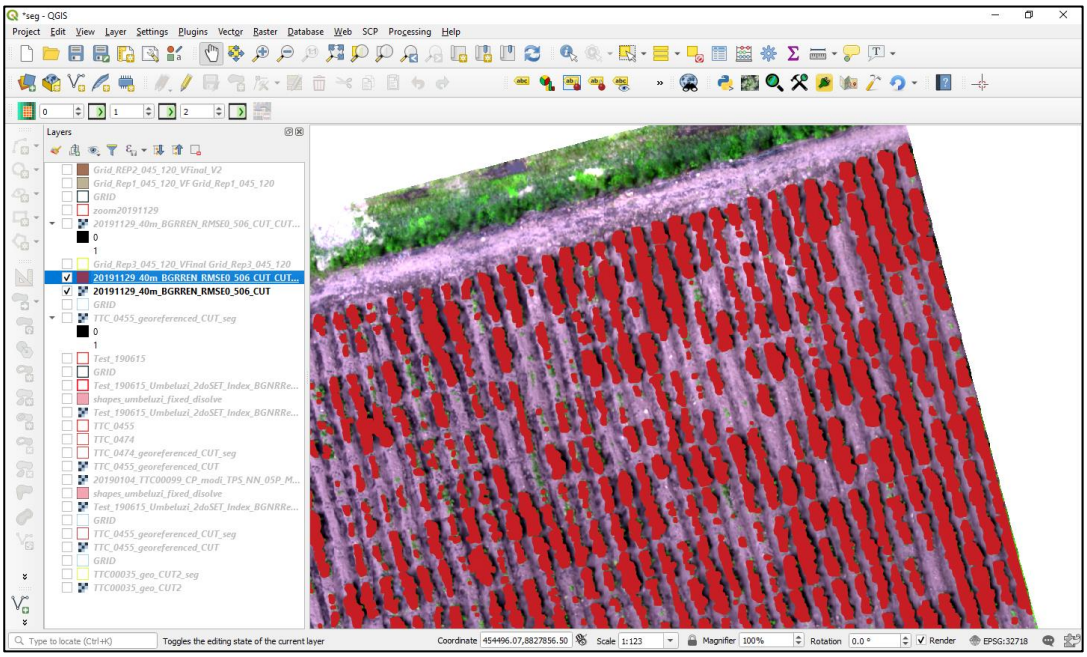

Fig. 20

9. To activate the plot labels, click on the icon  from QGIS, then a window will open on the right side as shown in Fig. 21.

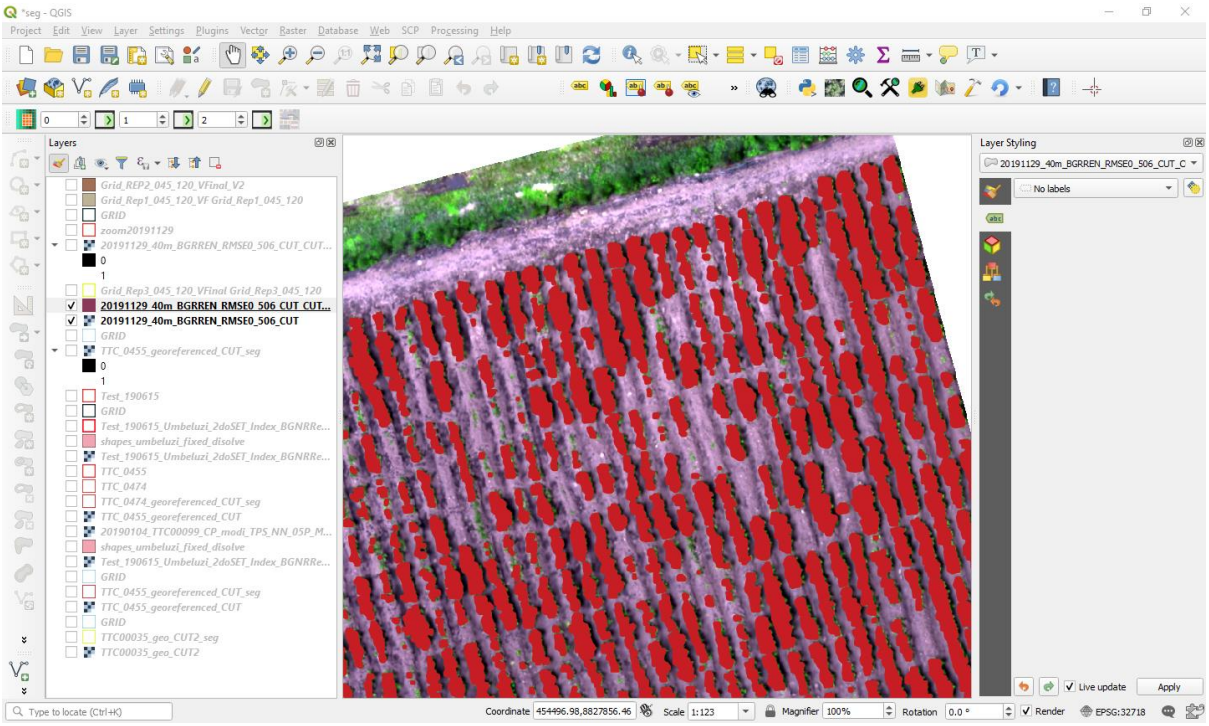

Fig. 21

10. The .shp file must be selected to see the labels in Fig. 22, in addition, the 'single lables' option must be activated as in Fig. 23.

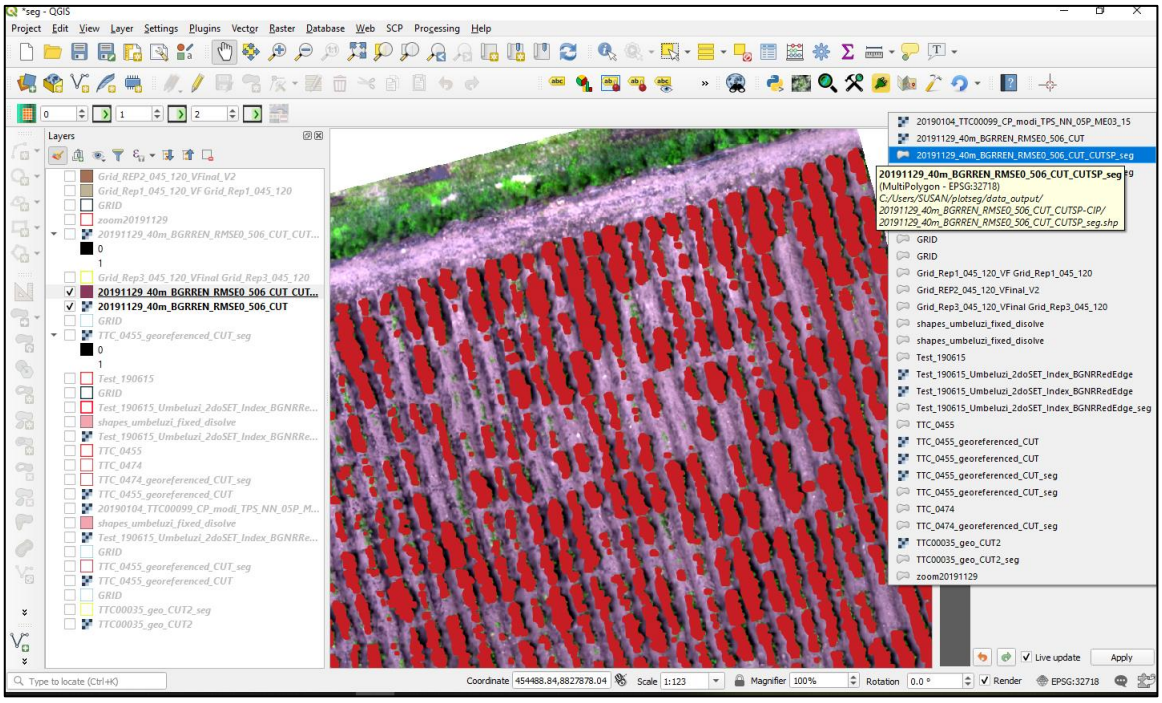

Fig. 22

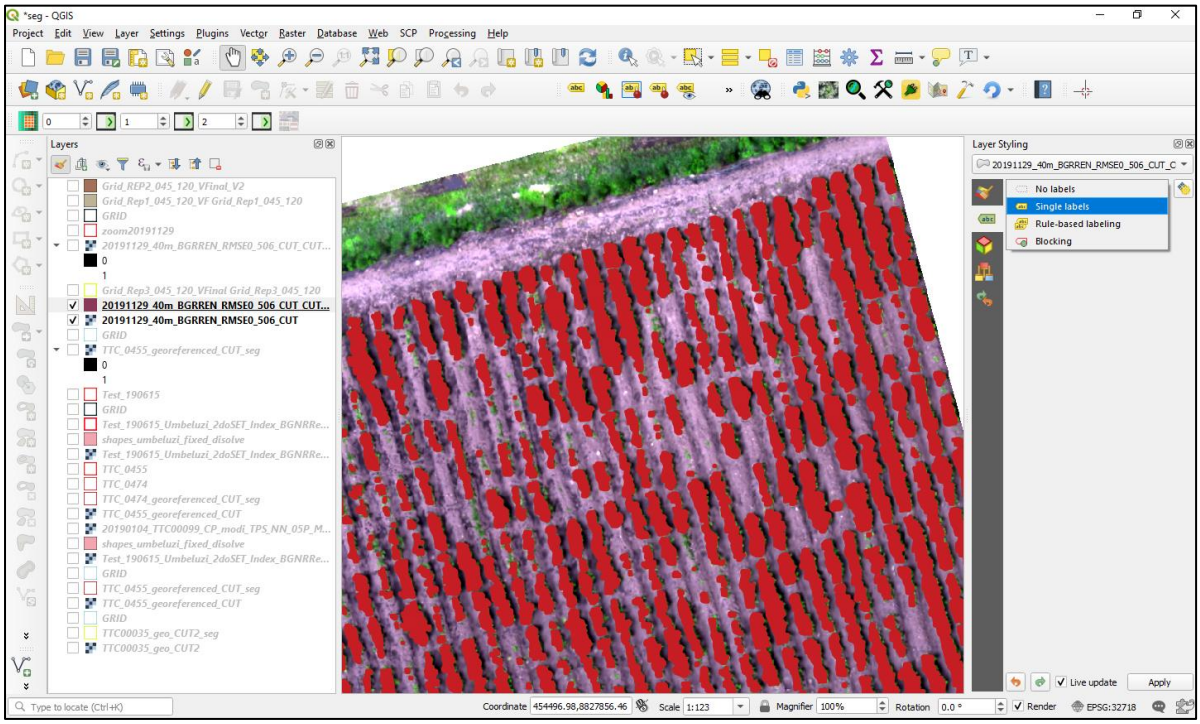

Fig. 23

11. It is observed that the labels will appear in the image. The size and color of the labels can be changed as indicated Fig.24.

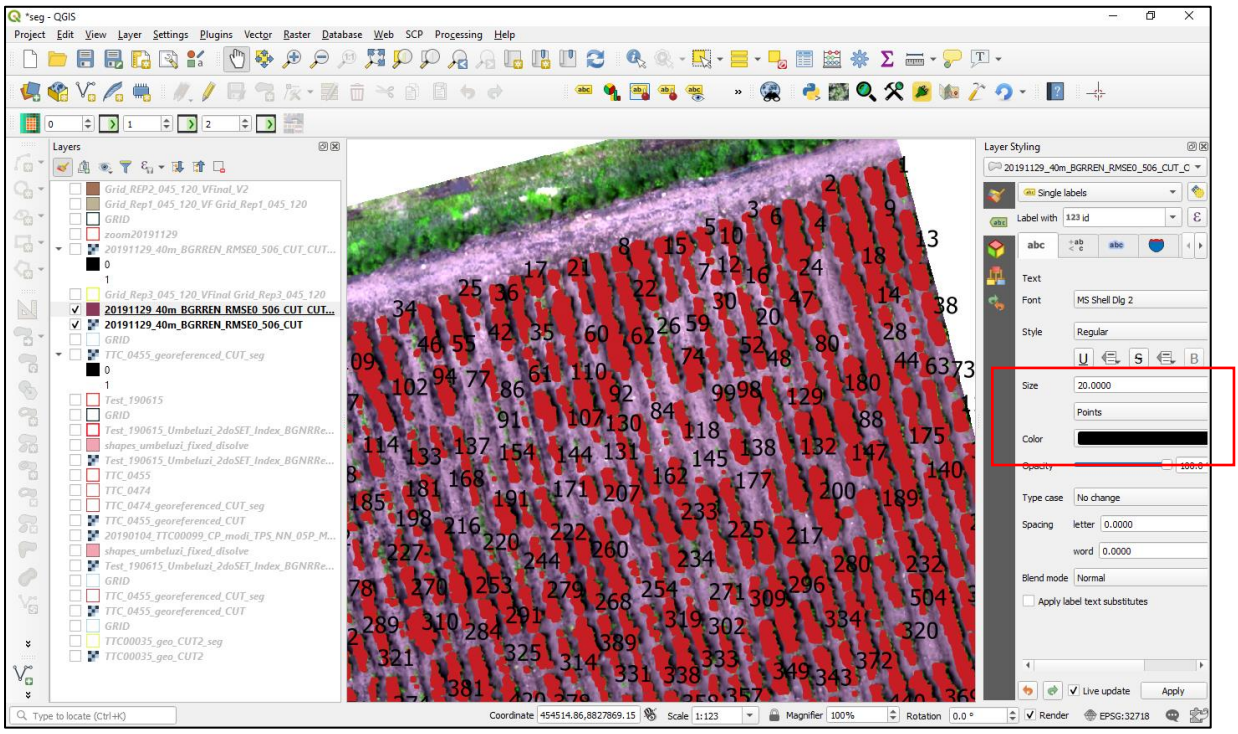

Fig. 24

12. Also the binary image can be loaded in QGIS (Fig. 25).

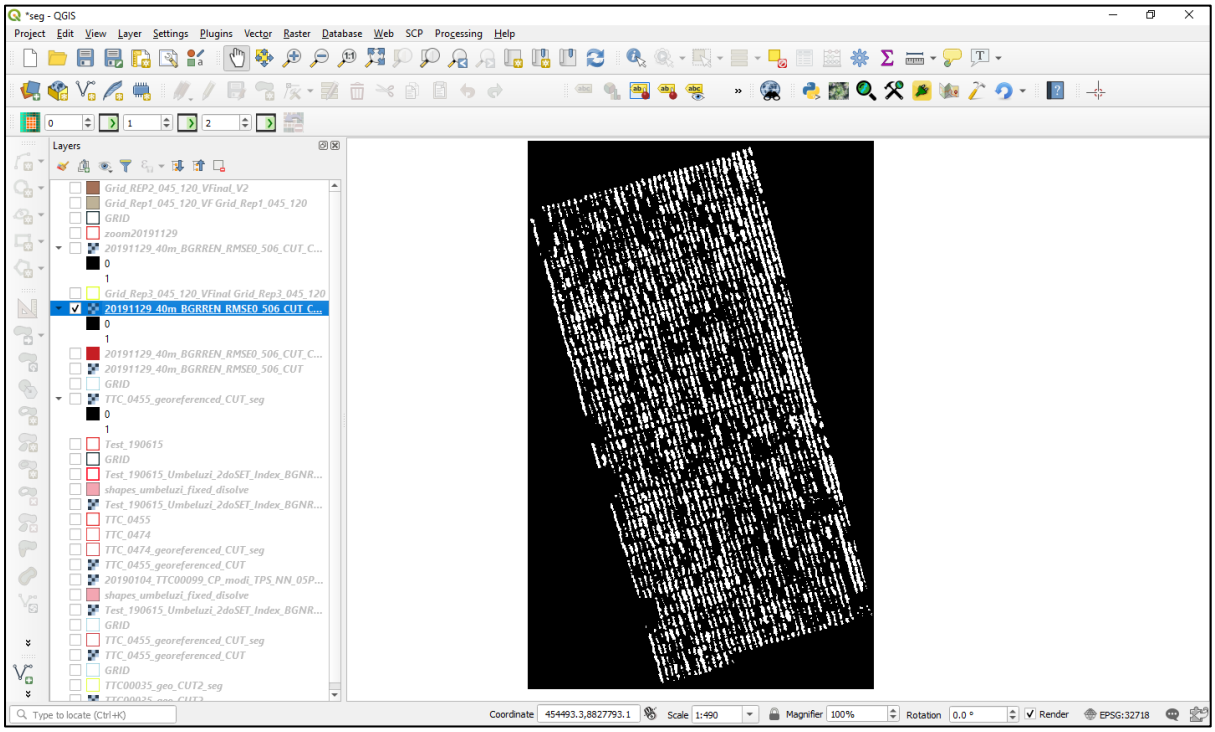

Fig. 25

13. To deactivate the variable in Anaconda write the command (Fig. 26)

```
conda deactivate
```

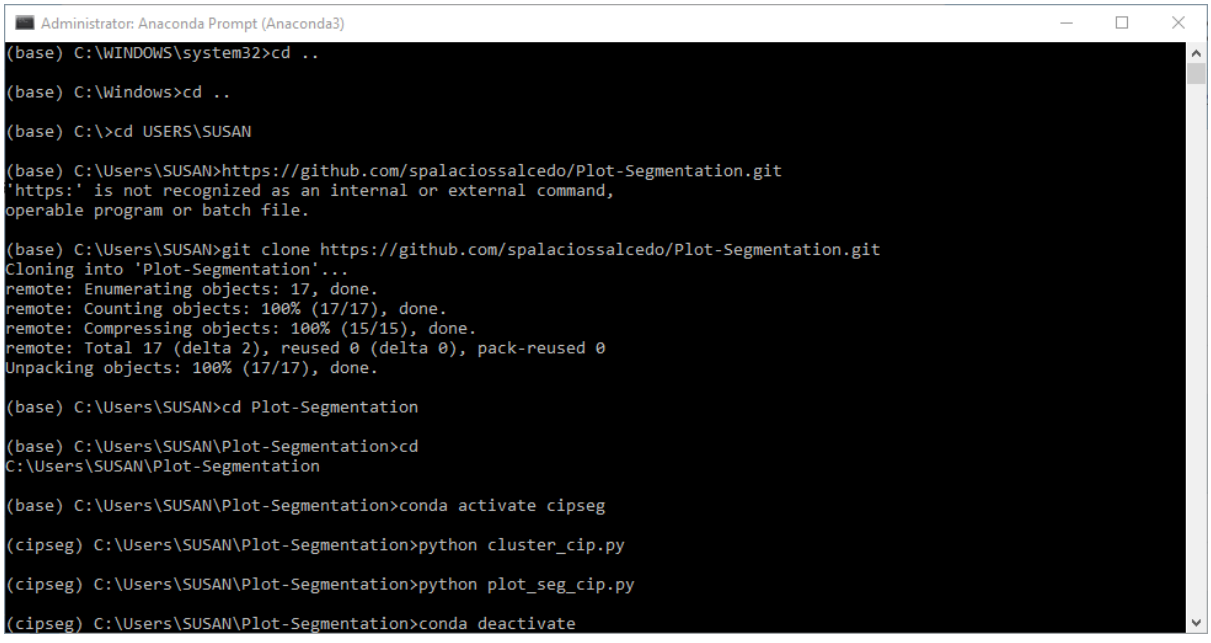

Fig. 26

14. To exit the command line, type exit and enter (Fig. 27).

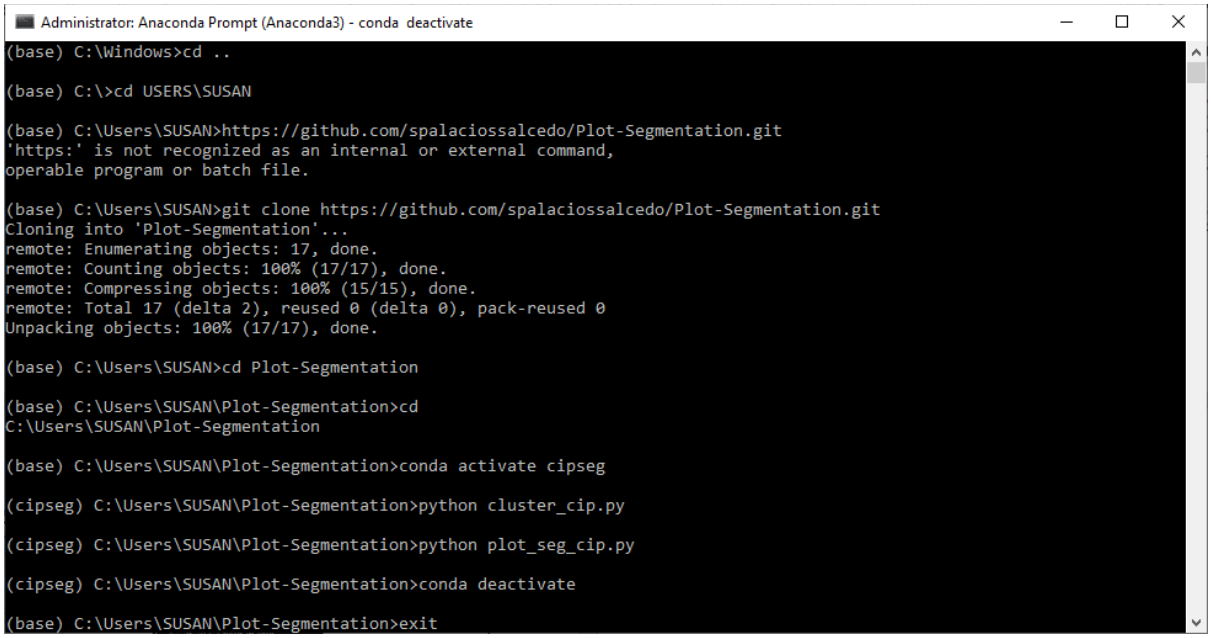

Fig. 27