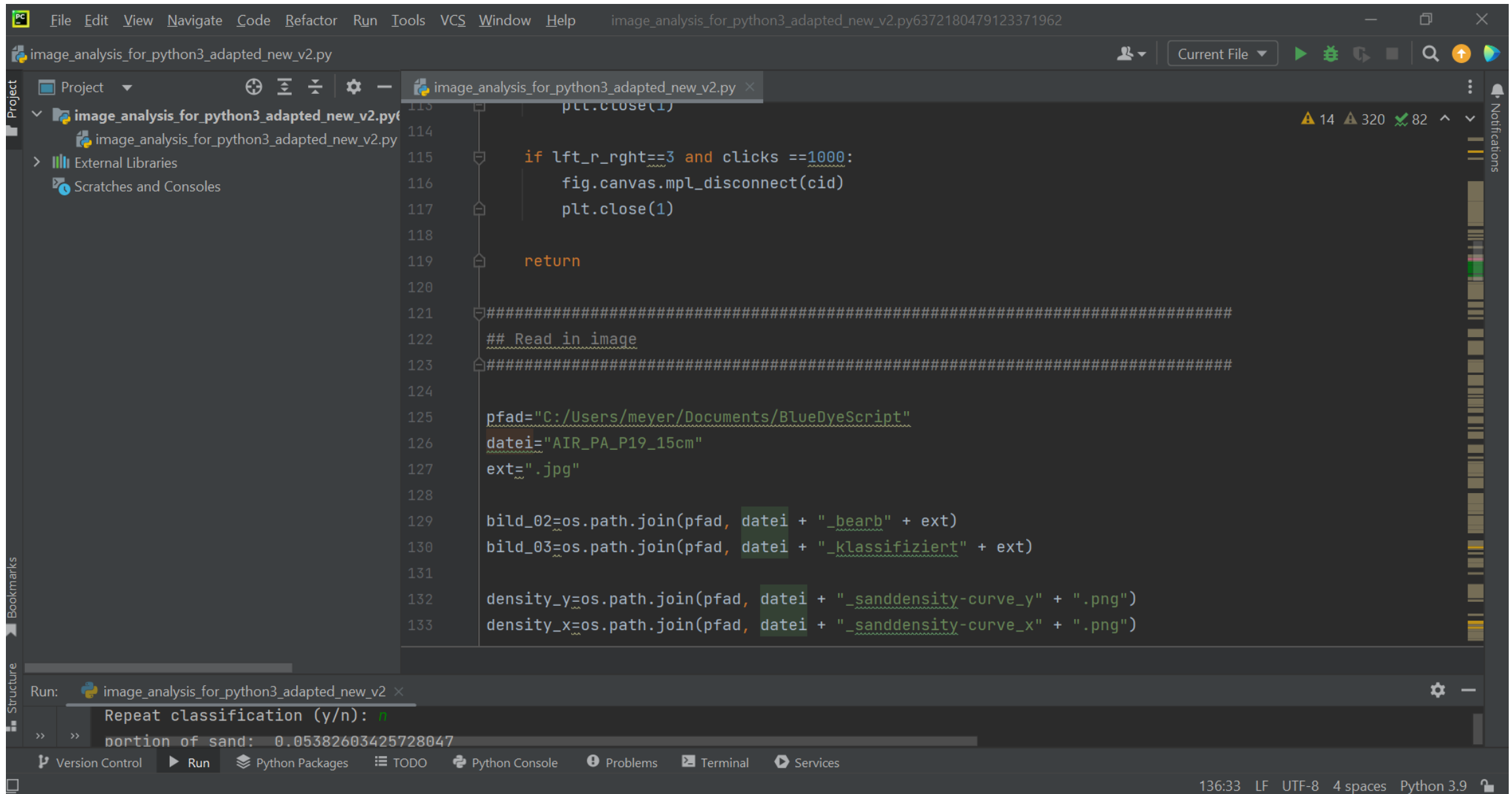
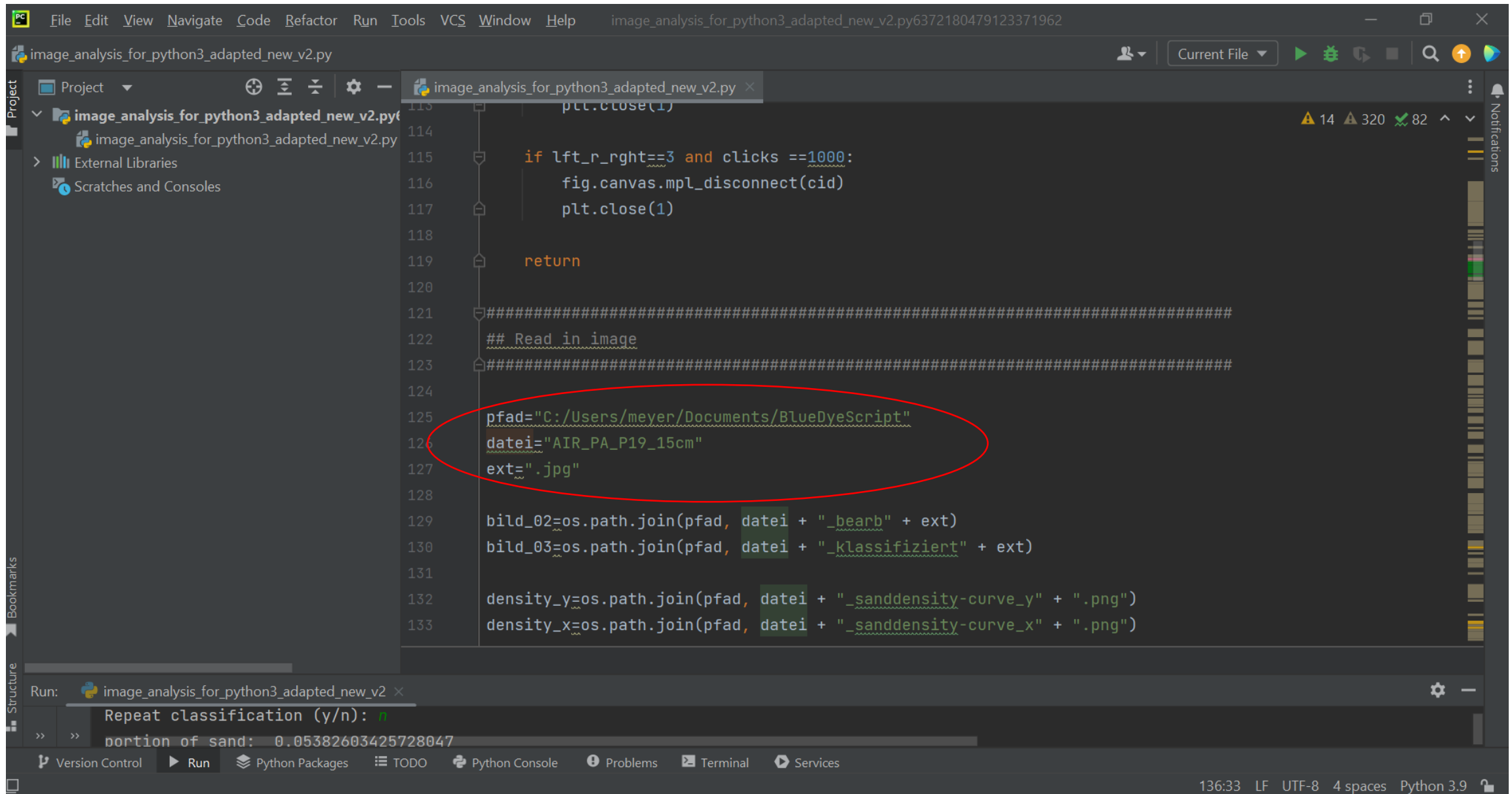


Pycharm interface with the blue dye script open



Update path, file name and extension



The screenshot shows a code editor with a Python file named `image_analysis_for_python3_adapted_new_v2.py`. The code is as follows:

```
113 plt.close(1)
114
115 if lft_r_rght==3 and clicks ==1000:
116     fig.canvas.mpl_disconnect(cid)
117     plt.close(1)
118
119 return
120
121 #####
122 ## Read in image
123 #####
124
125 pfd="C:/Users/meyer/Documents/BlueDyeScript"
126 datei="AIR_PA_P19_15cm"
127 ext=".jpg"
128
129 bild_02=os.path.join(pfd, datei + "_bearb" + ext)
130 bild_03=os.path.join(pfd, datei + "_klassifiziert" + ext)
131
132 density_y=os.path.join(pfd, datei + "_saddensity-curve_y" + ".png")
133 density_x=os.path.join(pfd, datei + "_saddensity-curve_x" + ".png")
```

A red circle highlights the variables `pfd`, `datei`, and `ext` on lines 125, 126, and 127 respectively. The IDE interface includes a menu bar at the top, a sidebar on the left with 'Project', 'External Libraries', and 'Scratches and Consoles', and a bottom panel with a 'Run' section showing the command prompt output: 'Repeat classification (y/n): n' and 'portion of sand: 0.05382603425728047'.

Click Play

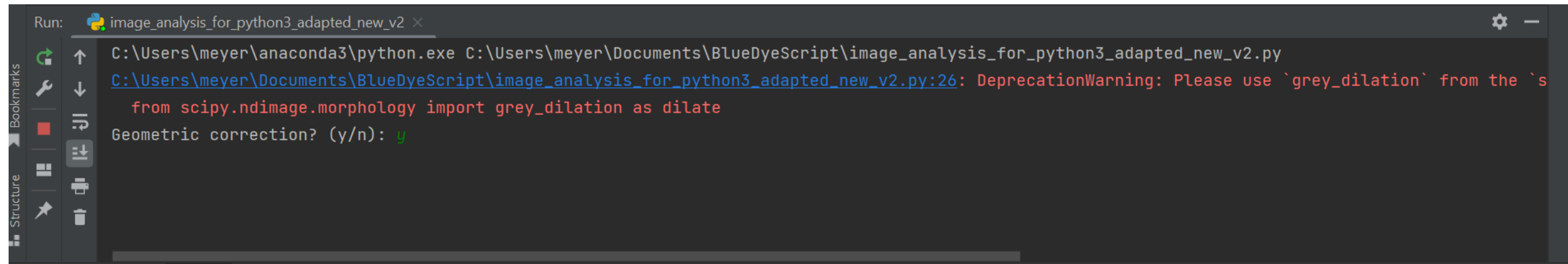
The image shows a screenshot of an IDE (likely PyCharm) with a dark theme. The main editor window displays a Python script named `image_analysis_for_python3_adapted_new_v2.py`. The script contains several lines of code, including a loop that checks for a specific number of clicks and a section that reads in an image and saves it. The code is as follows:

```
113 plt.close(1)
114
115 if lft_r_rght==3 and clicks ==1000:
116     fig.canvas.mpl_disconnect(cid)
117     plt.close(1)
118
119 return
120
121 #####
122 ## Read in image
123 #####
124
125 pfad="C:/Users/meyer/Documents/BlueDyeScript"
126 datei="AIR_PA_P19_15cm"
127 ext=".jpg"
128
129 bild_02=os.path.join(pfad, datei + "_bearb" + ext)
130 bild_03=os.path.join(pfad, datei + "_klassifiziert" + ext)
131
132 density_y=os.path.join(pfad, datei + "_saddensity-curve_y" + ".png")
133 density_x=os.path.join(pfad, datei + "_saddensity-curve_x" + ".png")
```

The IDE interface includes a menu bar at the top with options like File, Edit, View, Navigate, Code, Refactor, Run, Tools, VCS, Window, and Help. The top toolbar shows a 'Run' button (a green play icon) which is circled in red. The left sidebar contains a 'Project' view showing the file structure and a 'Run' view at the bottom showing the execution output. The 'Run' view displays the following text:

```
Run: image_analysis_for_python3_adapted_new_v2
Repeat classification (y/n): n
portion of sand: 0.05382603425728047
```

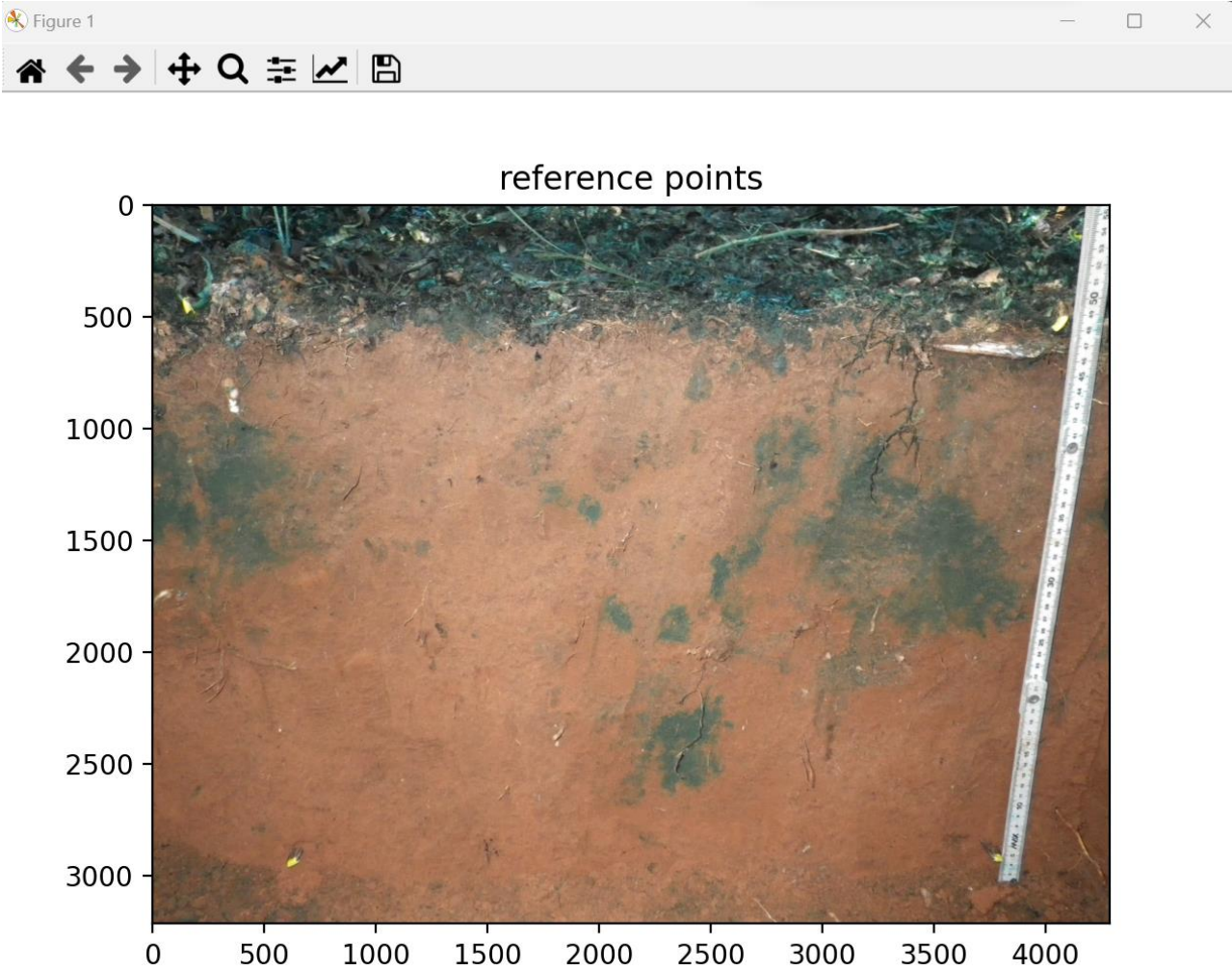
The bottom status bar indicates the current line is 136:33, the file encoding is UTF-8, and the Python version is 3.9.



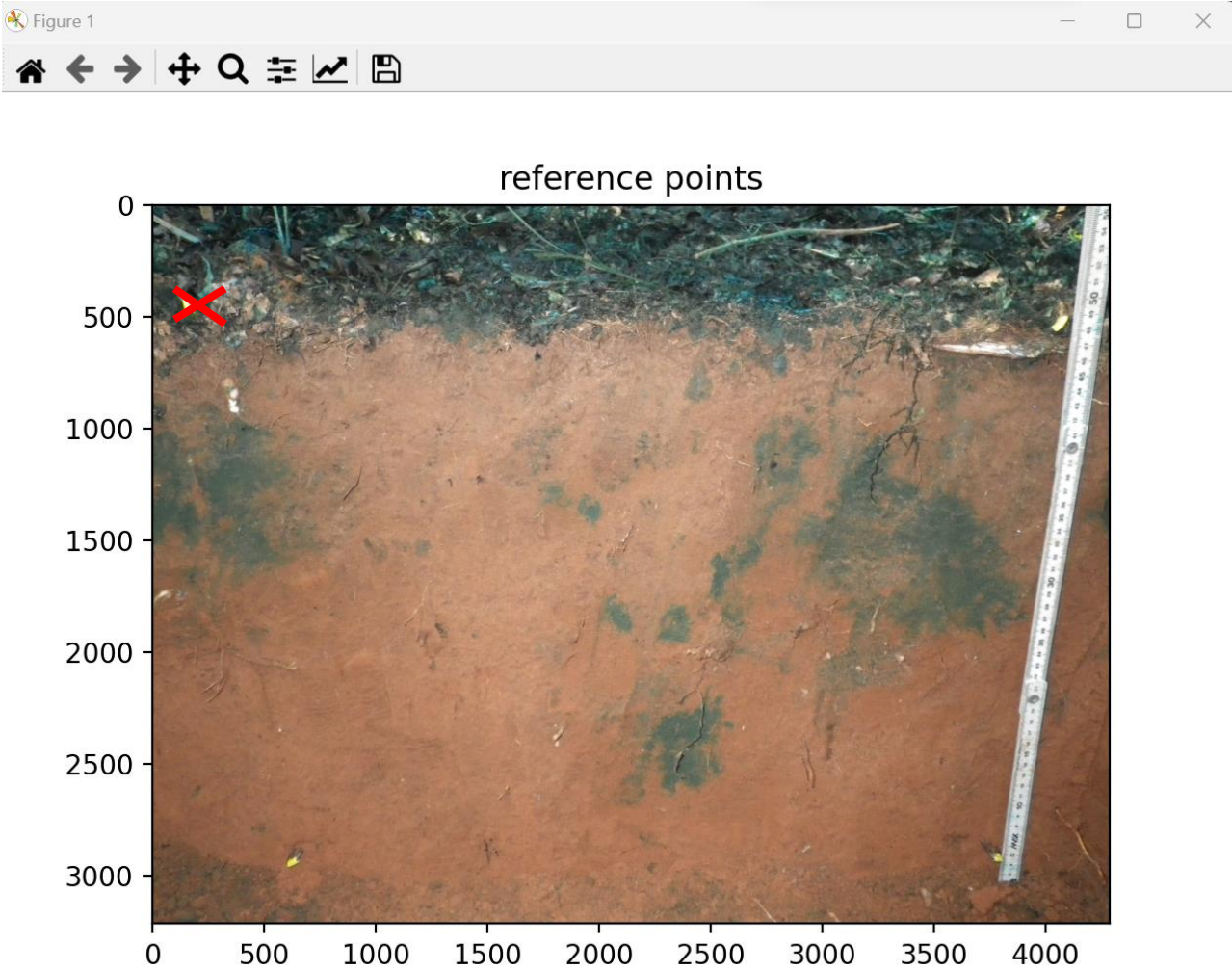
```
Run: image_analysis_for_python3_adapted_new_v2 ×  
C:\Users\meyer\anaconda3\python.exe C:\Users\meyer\Documents\BlueDyeScript\image_analysis_for_python3_adapted_new_v2.py  
C:\Users\meyer\Documents\BlueDyeScript\image_analysis_for_python3_adapted_new_v2.py:26: DeprecationWarning: Please use `grey_dilation` from the `s  
    from scipy.ndimage.morphology import grey_dilation as dilate  
Geometric correction? (y/n): y
```

Geometric correction? Write “y”

This window pops up



Select with the mouse the first point

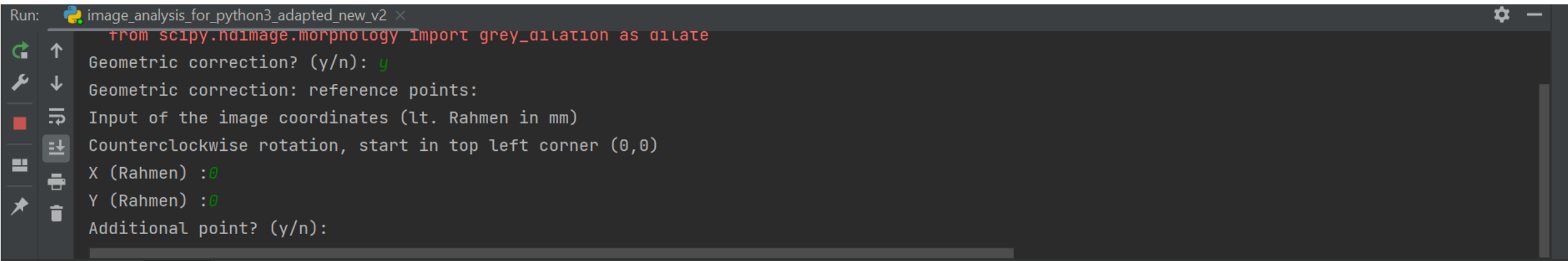


Fill in the coordinates of the first point (in millimeters):

X: 0

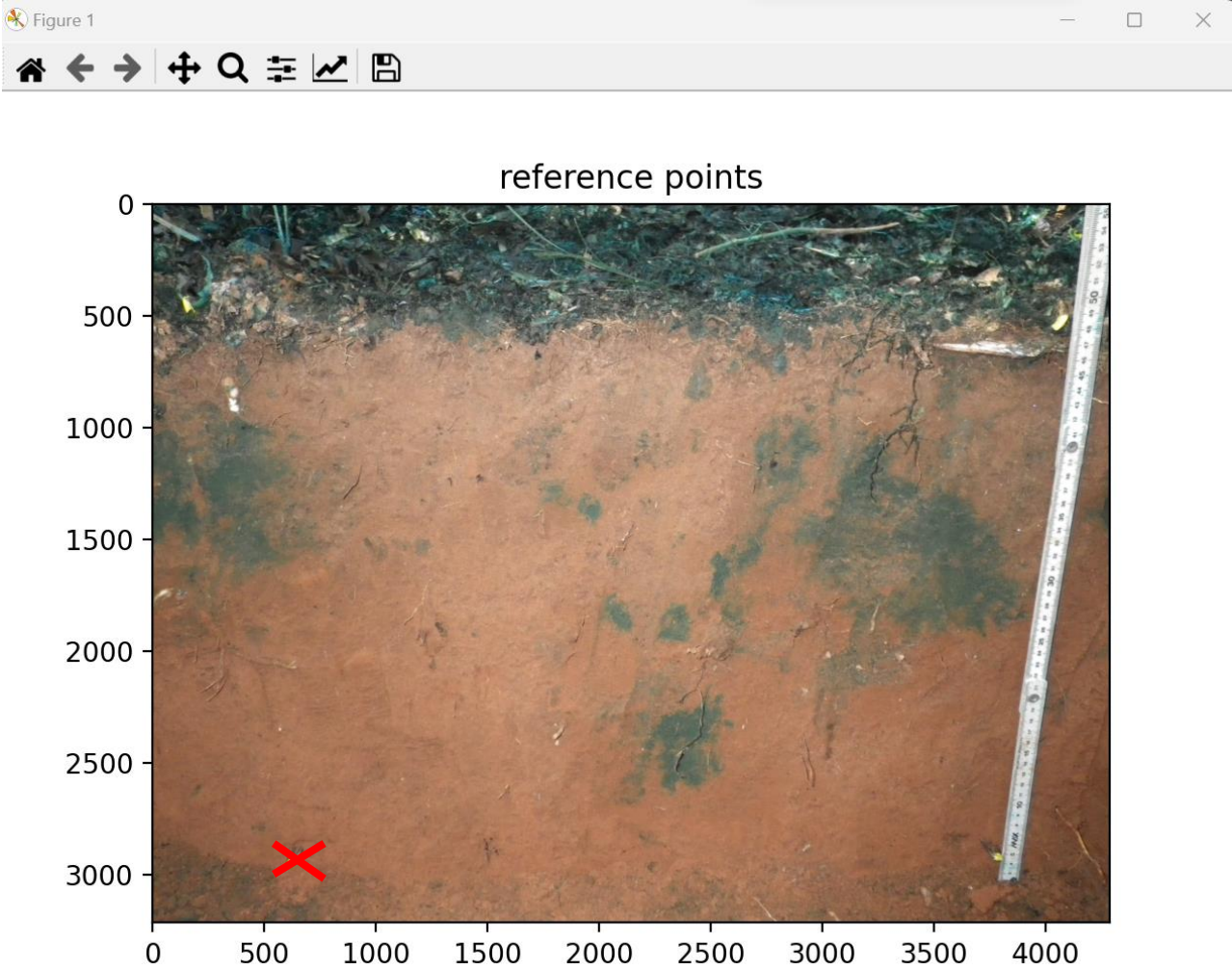
Y: 0

Additional point? Type "y"



```
Run: image_analysis_for_python3_adapted_new_v2 x
from scipy.ndimage.morphology import grey_dilation as dilate
Geometric correction? (y/n): y
Geometric correction: reference points:
Input of the image coordinates (lt. Rahmen in mm)
Counterclockwise rotation, start in top left corner (0,0)
X (Rahmen) :0
Y (Rahmen) :0
Additional point? (y/n):
```


The photo pops up again
Select the other point



Fill in the coordinates of the second point (in milimeters):

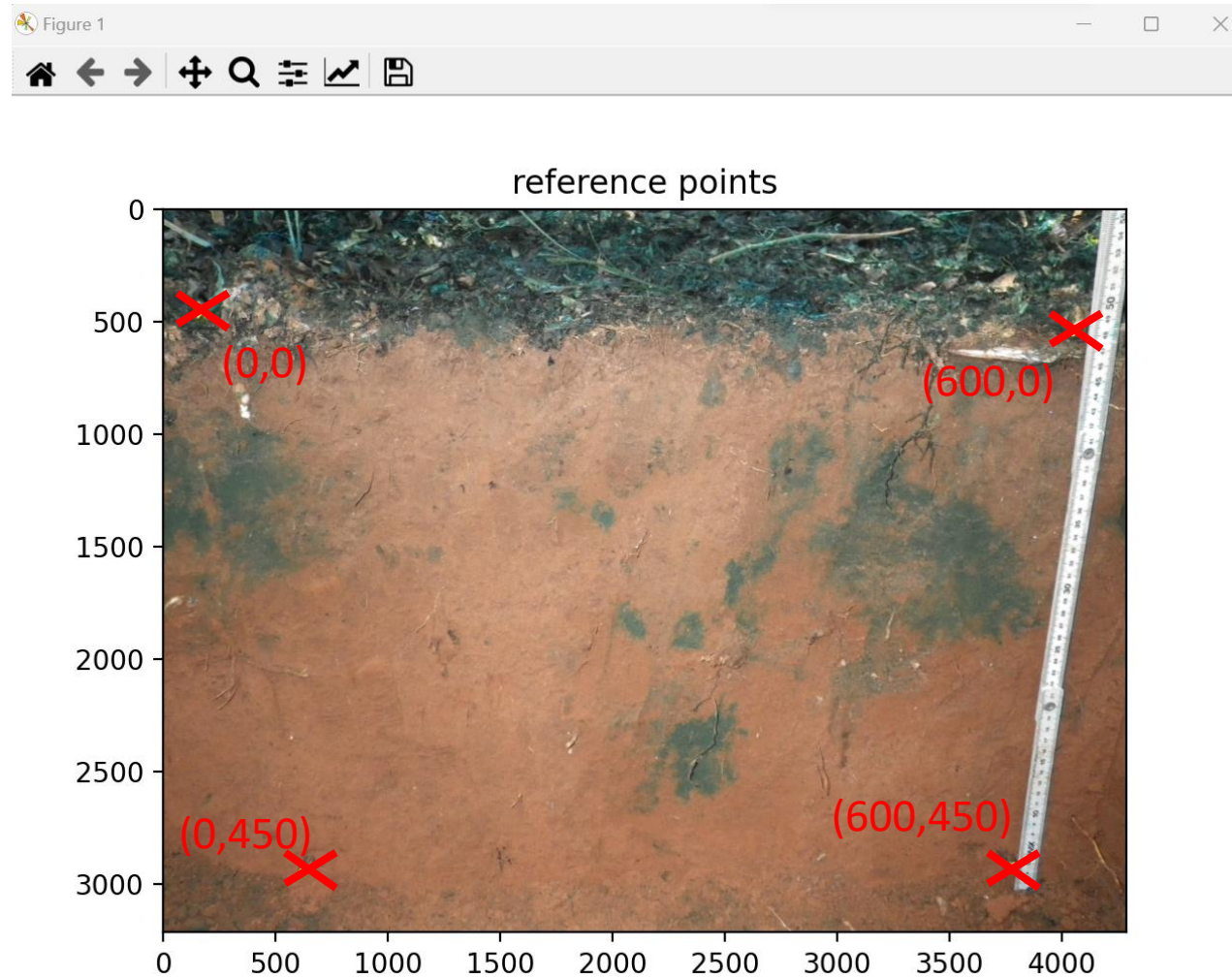
X: 0

Y: 450

Additional point? Type "y"

```
Run: image_analysis_for_python3_adapted_new_v2 x
Y (Rahmen) :0
Additional point? (y/n): y
Geometric correction: reference points:
Input of the image coordinates (lt. Rahmen in mm)
Counterclockwise rotation, start in top left corner (0,0)
X (Rahmen) :0
Y (Rahmen) :450
Additional point? (y/n):
```

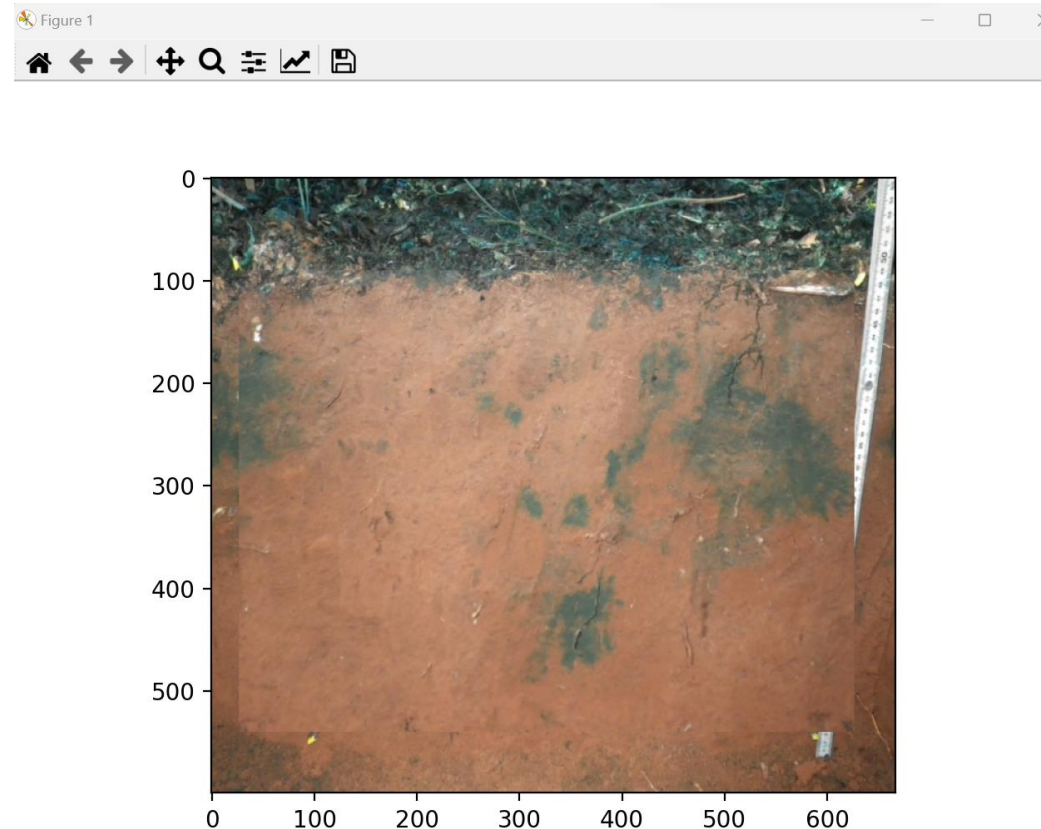
Do the same for the four points



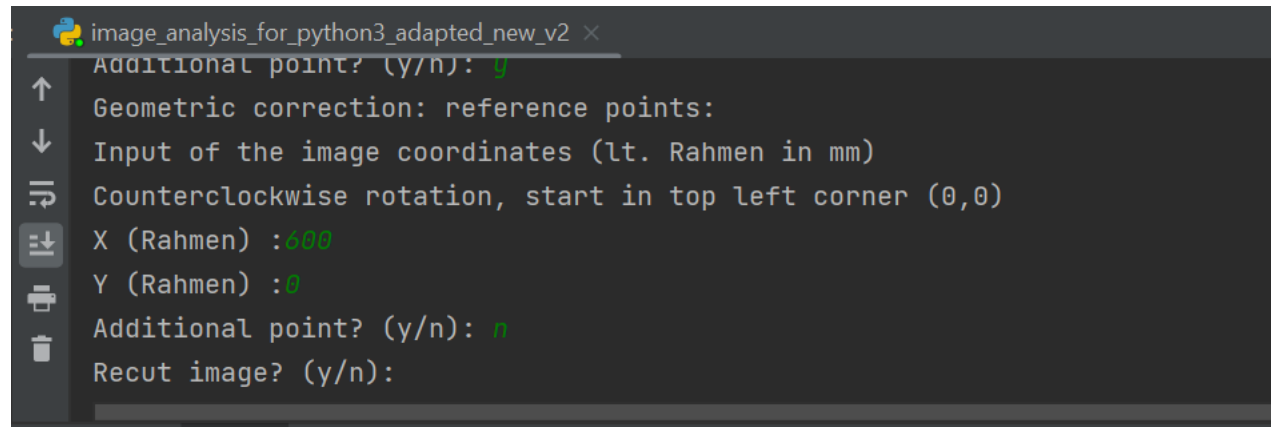
In my case, I had measured in the field that the frame was 60cm x 45 cm

When finished, Additional point? “n” for no

- The geometrically corrected figure window will pop up. Check it, and close the window. Go back to the script

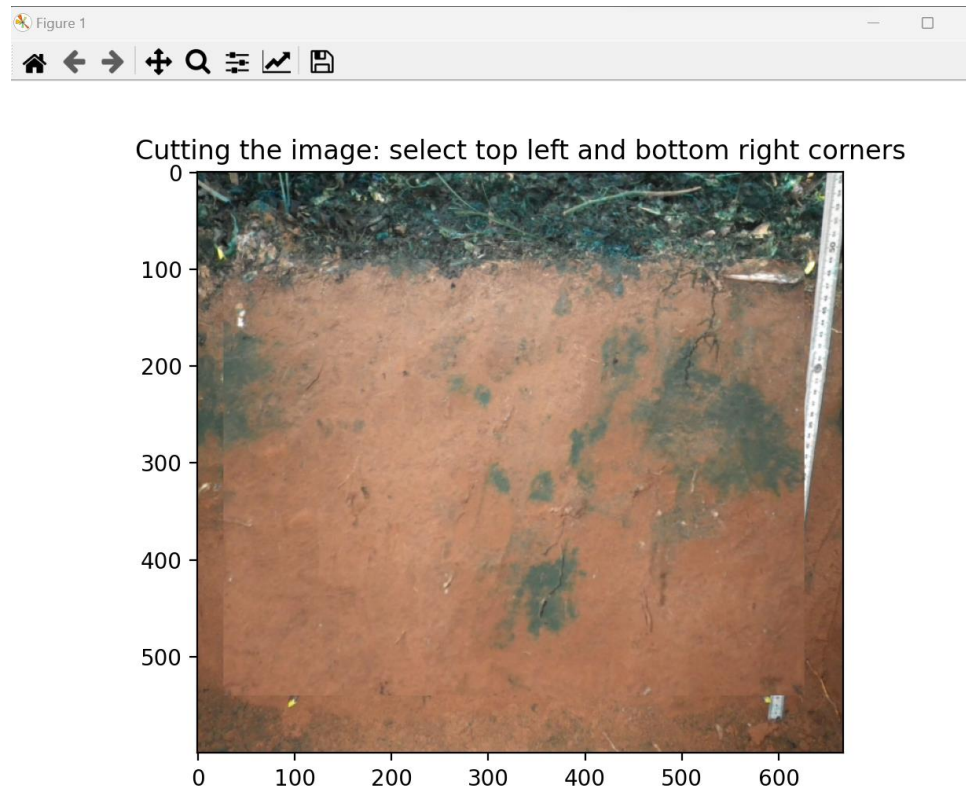


Recut image? Select “y”

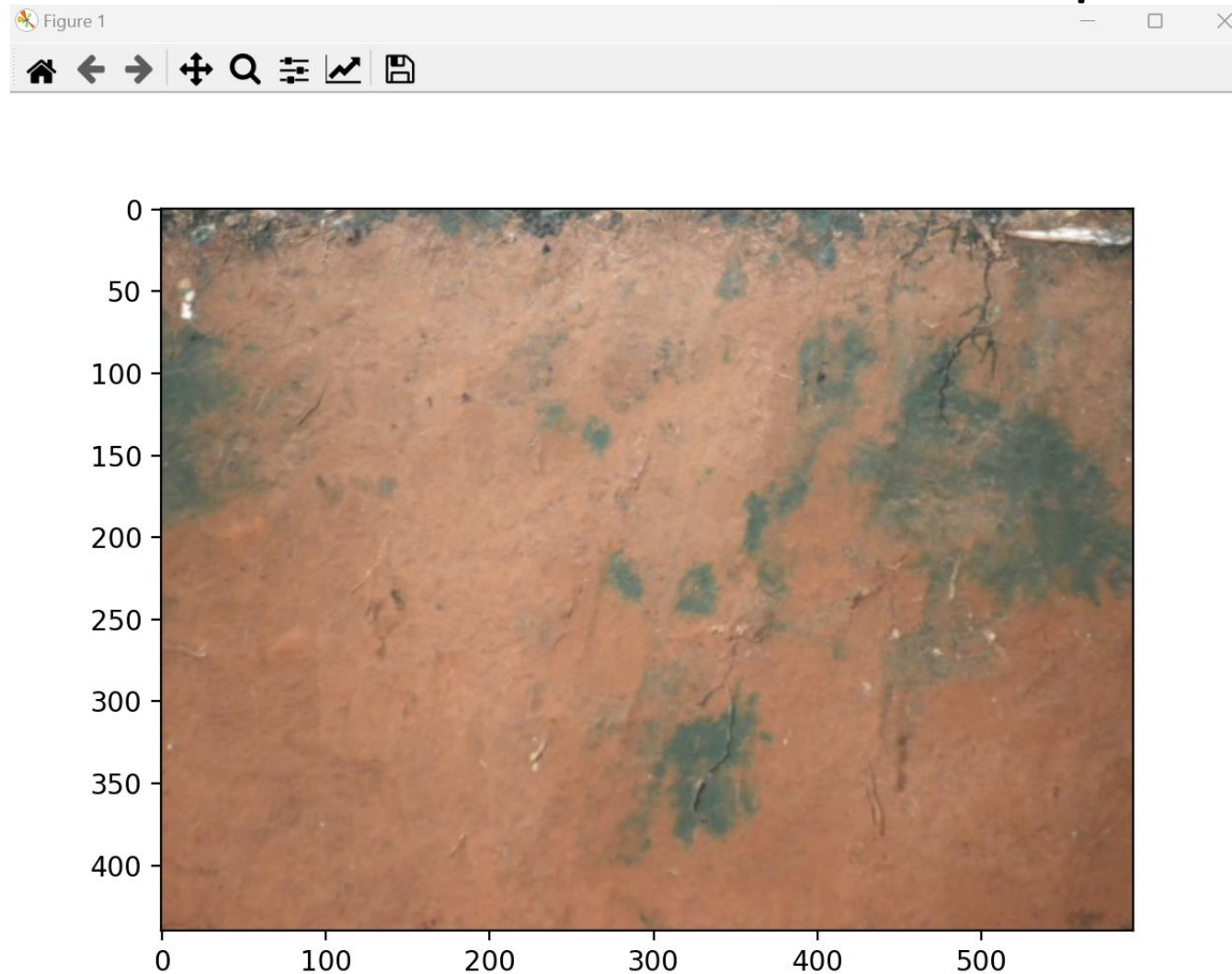


```
image_analysis_for_python3_adapted_new_v2 x
Additional point? (y/n): y
Geometric correction: reference points:
Input of the image coordinates (lt. Rahmen in mm)
Counterclockwise rotation, start in top left corner (0,0)
X (Rahmen) :600
Y (Rahmen) :0
Additional point? (y/n): n
Recut image? (y/n):
```

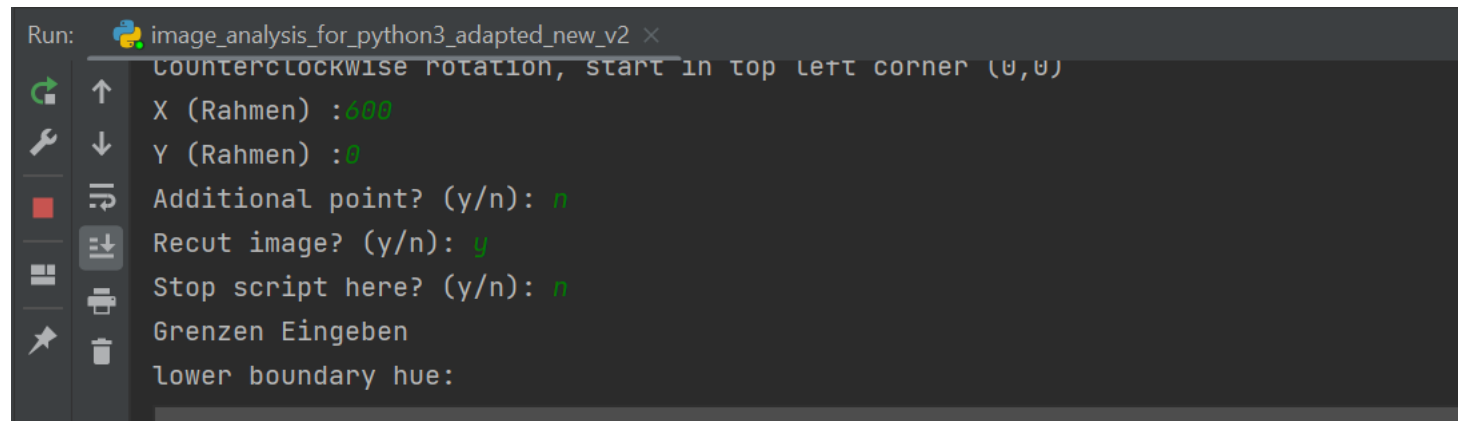
This window will pop up, then just select top left and bottom right corners



The cut figure will pop up. Check it and close the window. Go back to the script



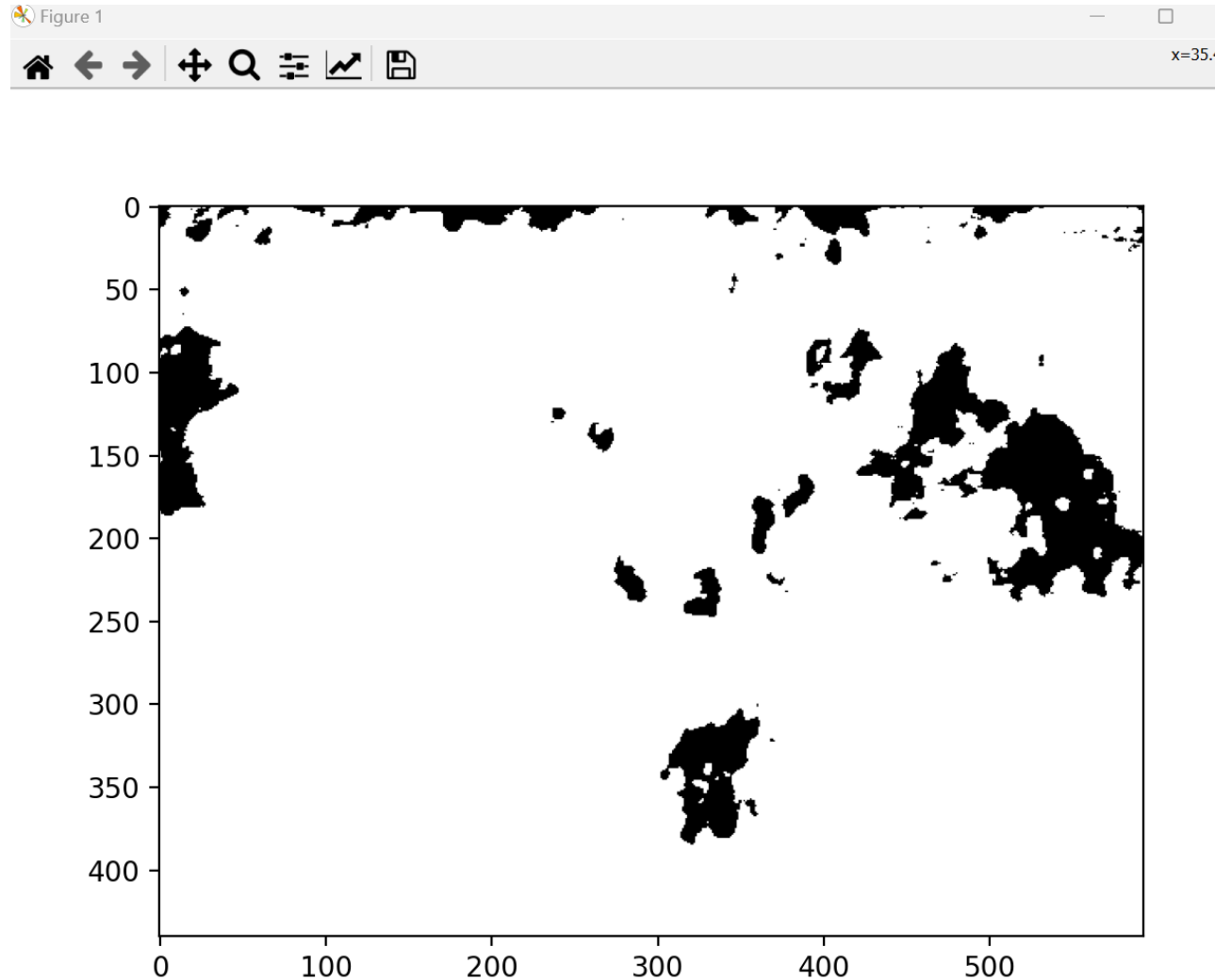
Stop script here? No
Then fill in:
lower boundary hue
upper boundary hue
lower boundary val
upper boundary val



```
Run: image_analysis_for_python3_adapted_new_v2 x
CounterClockwise rotation, start in top left corner (0,0)
X (Rahmen) :600
Y (Rahmen) :0
Additional point? (y/n): n
Recut image? (y/n): y
Stop script here? (y/n): n
Grenzen Eingeben
lower boundary hue:
```

Values I used:
lower boundary hue: 50
upper boundary hue: 200
lower boundary val: 0
upper boundary val: 0

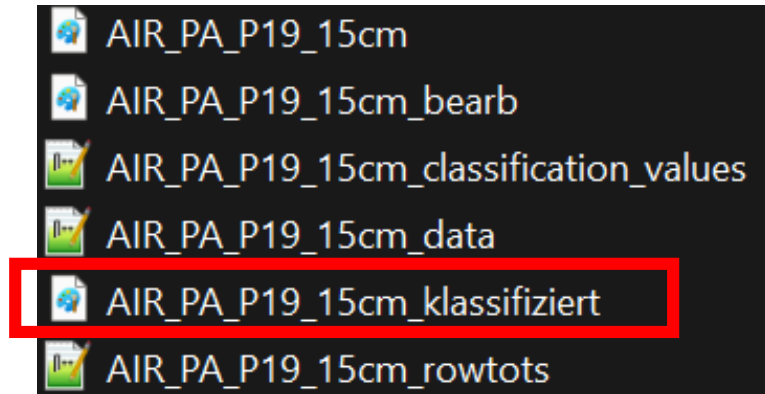
The classified figure will pop up. Check it, close the window. Go back to the script



```
Run: image_analysis_for_python3_adapted_new_v2 x
Grenzen Eingeben
lower boundary hue: 50
upper boundary hue: 200
lower boundary val: 0
lower boundary y: 0
portion of sand: 0.07721140579487966
portion of not sand: 0.9227885942051204
Repeat classification (y/n):
```

- If you liked the classification, “repeat classification -> no”
- If you want to try again with other boundaries, “repeat classification -> yes”
- “Portion of sand”: pixels classified as blue -> 7.7%

During the process, it has generated some files in the folder specified. The classified figure is there.



Here's what the "data" file looks like:

[illegible]

You can open it in Excel (values separated by comma) to double-check:

sum	total	ratio				
20146	260920	0.077211				
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	0
1	1	1	1	1	0	0
1	1	1	1	1	1	0
1	1	1	1	1	1	1
1	1	1	1	1	1	0
1	1	1	1	1	1	1
1	1	1	1	1	0	0
1	1	1	1	0	0	0
1	1	1	1	0	0	0
1	1	1	0	0	0	0
1	0	0	0	0	0	0
0	0	0	0	0	0	0