Sasha Ries 4/29/23

**Instructions for using the Automatic Image Analyzer and User Interface**

1. Place all images to be analyzed in the folder called “Images\_in” which will be located in the same directory (folder location) the program is in.

-The images must follow the naming scheme: “sampleType\_testType\_minLoad\_maxLoad\_#N\_#Cycles\_back.bmp”. Number of cycles and back are additional things depending on if the image is a back image or cycle number is specified. If they are not included just omit them.

1. If you have a text editor for code (Visual Studio Code) then you will be able to directly edit the code and run the program from there by clicking the run button (small triangle in the top right corner)

**To access via command window through jumbo folder:**

-On Windows, press the Windows key + R to open the Run dialog box, type "cmd" and press Enter.

-On macOS, press Command + Space to open Spotlight search, type "terminal" and press Enter.

-In the command window type: pushd + file directory (S:\orthopedics\PCS\Sasha\_automating). Next run the file by typing the **py +** full name of program (**py analysis\_automated.py** for example).

**Note:** Since images are being accessed from the jumbo folder the program will run much slower. A way around this is to download a copy of the python files and images into a different folder on your computer disc and run it from there

-To close the program before its finished type **Command + c** in the command window or click on the interface window and **press the s key**.

**To access via command window through directory on computer disc:**

-On Windows, press the Windows key + R to open the Run dialog box, type "cmd" and press Enter.

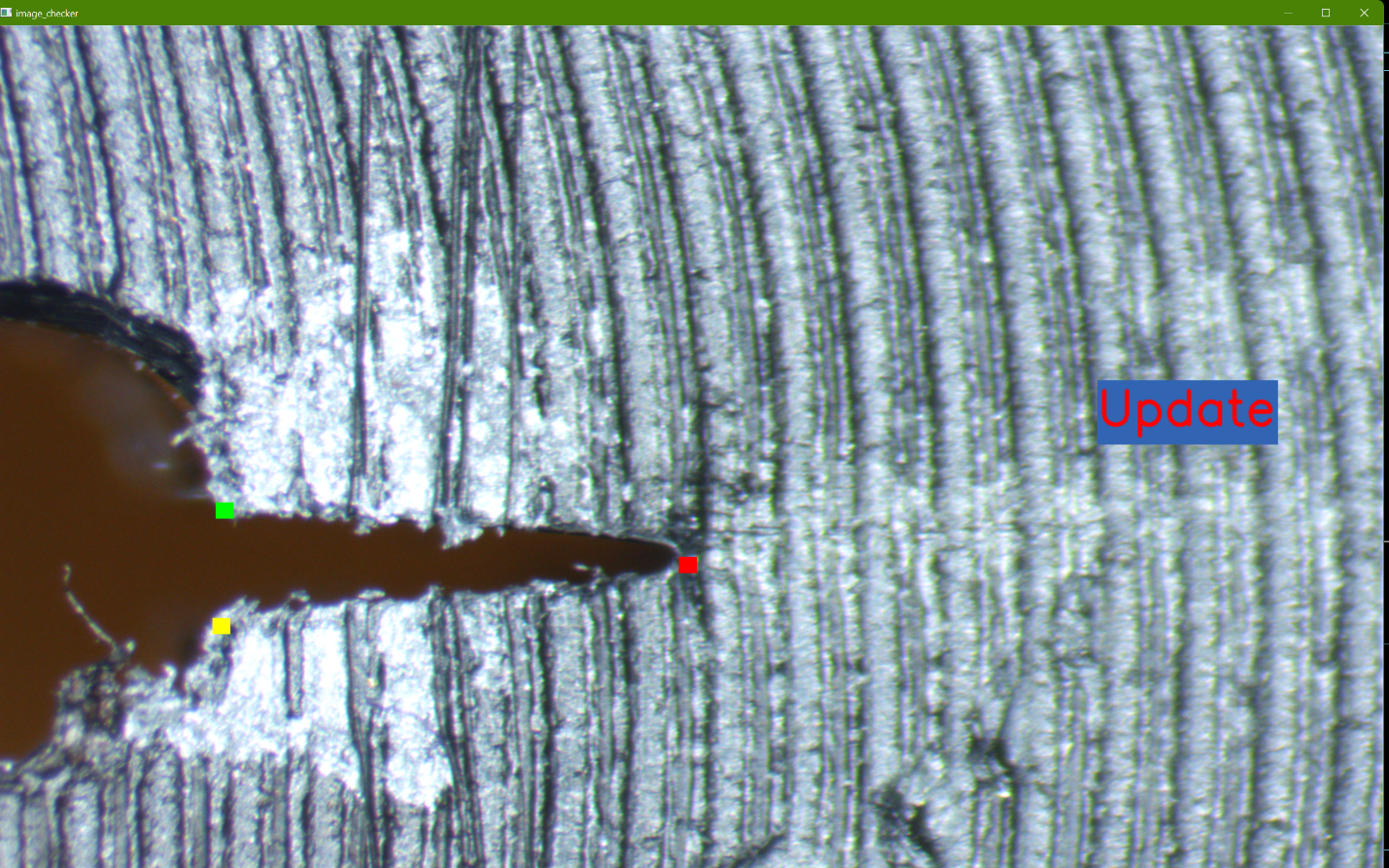
-On macOS, press Command + Space to open Spotlight search, type "terminal" and press Enter.

-In the command window type: cd + file directory (C:\folder1\folder2\program\_folder). Next run the file by typing the **py +** full name (**py analysis\_automated.py** for example).

**3.** Once the images are analyzed you can then run the image\_checker.py file which will

set up the interface to let you check and adjust where the locations were found.

1. Run that program the same was as specified above then click on the squares to move

them to desired location and click again to leave it there. **Green square is top corner of the crack, Yellow square is bottom corner, and Red square is crack tip**. Then either click space or the update button to update the location in the excel file and move to the next image. Once they are in the correct location it will look as follows (top left corner of the square corresponds to coordinates):

**Note:** while either program is running the excel file must be closed, otherwise it won’t be able to access it and update the data. Sheet 1 will be overwritten with data every time it is updated but sheet 2 will remain unaffected.

1. Now your done! The last two columns of the excel Sheet1 are the horizontal distance between the averaged corner location and the crack tip in pixels.