**Step I** Training data preparing (Settings should be the same in the whole study)

1. using fluorescence microscopy to take images with three channel of plant tissues, and two channels of pure bacteria. Each channel should have no leakage to other channels. In the example, tdTomato, GFP and autofluorescence for plants; tdTomato and GFP for bacteria. The resolution of the images should be 512x512.
2. using ImageJ/Fiji to split the training images into different channels and put them in the "training\_dataset/bacteria/" and "training\_dataset/plant\_tissue/" as TIFF files. For example, Image.czi of roots was splitted into C1-Image.tif (tdTomato), C2-Image.tif (GFP) and C3-Image.tif (AutoFluorescence). C1-Image.tif should be in "training\_dataset/plant\_tissue/tdTomato/". C2-Image.tif in "training\_dataset/plant\_tissue/GFP/". C3-Image.tif in "training\_dataset/plant\_tissue/Autofluorescence/".

**Step II** Input data preparing (one image (three channels) per time)

1. using fluorescence microscopy to take images with three channel of the sample. The resolution the images should be 512x512.
2. using ImageJ/Fiji to split the training images into different channels and put them in the "input/" sub-folders according to their fluorescence as TIFF files. For example, Image.czi of bacteria-colonized roots was splitted into C1-Image.tif (tdTomato), C2-Image.tif (GFP) and C3-Image.tif (AutoFluorescence). C1-Image.tif should be in "input/tdTomato/". C2-Image.tif in "input/GFP/". C3-Image.tif in "input/Autofluorescence/".

**Step III** run the program for processing

1. In linux and MacOS the python3 are pre-installed. If you are using WindowsOS, please install python3 (64-bit) before installing the python packages (website: https://www.python.org/). First download the python3.exe installer and double click it. Before finishing the installing, the Installer will ask whether add it to the PATH, please allow it!
2. Before running the program, please make sure the following python3 (64-bit) packages are installed in your environment. If not, just input the following commond line in the terminal and run it "pip3 install torch torchvision torchaudio numpy scikit-image matplotlib“
3. For Linux and MacOS, in the terminal input: "python3 Af\_Removal.py". Then the program will be executed (make sure the terminal is opened in the Af\_Removal folder or the it can not find the input and training data). For WindowsOS, just double click the Af\_Removal.py. You can also run the program in the windows terminal if you like, the command is the same. If the python3 is the only python you installed on windows, just input "python Af\_Removal.py" in the windows terminal.

**Step IV** input the parameters the program asks

The program will check whether you prepared the training data and input data properly. If everything seems fine, it will ask you for a few parameters, which may influence the how the output look like at the end. If you are not clear what to do, just used the suggested settings.

Be patient when the program is running, it takes some time for the machine to predict the background autofluorescence.

**Notes**

* Don’t worry if cuda is not available on your computer, the output with CPU will be same as that from a cuda available computer, but it may take longer time for processing.
* In the application there are already input and training data, you can try the performance of this approach with them. As a pre-trained model was used in this application ("cnn\_pretrained.plk"), 100~200 cycles of processing are enough to generate a good image.
* If you want to process your own image, please replace the existing data with yours. For the first time of training with new data, it will take some time to finish. We suggest to train the model for more than 1500 cycles for the first time with your own data, then save the trained model (the application will save it for you and inform you the saved file name) and used it as the pre-trained model next time for other images from your own study (delete the "cnn\_pretrained.plk" file, and replace it with the new model generated by you).
* Feel free to contact this email-address (jiangxun@uni-bremen.de) if you have any doubts, questions and suggestions on this approach and the written python program.