# DL_Track_US v0.2.1

# Preface

Welcome to the DL_Track_US python package tests. Because we have not (yet) integrated unit testing in the DL_Track_US python package, we have prepared specific instructions and provided example results. You will perform several analyses and compare your results to the test result we provided. If the results are comparable, the DL_Track_US python package is functional. Have fun!

**Please note that we updated the GUI from version 0.1.2 to version 0.2.1. Although it might look different, the core functionalities are the same. This is why we did not update the complete test tutorial. However, we refer to the new functionalities whenever necessary.**

# Content

# Good to know

All relevant instructions and guidelines for the installation of the DL_Track_US software package are described in our Github repository, so please take a look there if anything is unclear.

Before we start with testing, here are some important points to consider:

- Please **exactly** follow the instruction we provide!
- We assume that you have looked at the "DL_Track_US_tutorial.pdf" file prior to working through these test instructions. We therefore won't explain the core functionalities of the DL_Track_US GUI.
- All of the test files required for testing procedures are in the "DL_Track_US_example/tests" folder. If you have not downloaded the "DL_Track_US_example" folder, you can do so here. Unzip it and put it somewhere you can easily access it, you will use it extensively during testing.
- In case you encounter problems during the testing process or your test results deviate from the ones we provided without reasonable explanation, please report it. Take a look here how to best do this.

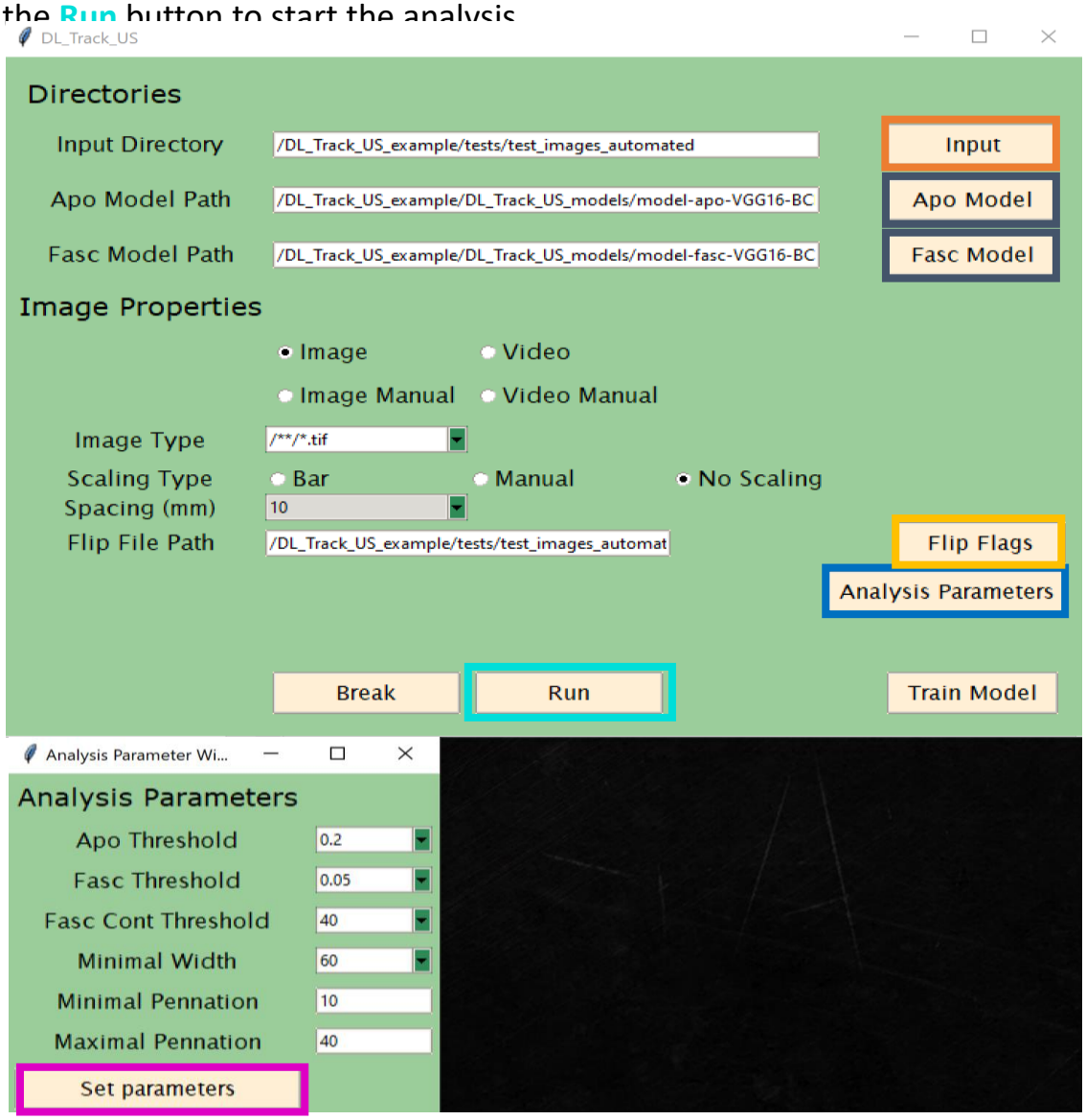
# Automated Image Analysis Test

The first analysis type you are going to test is the automated image analysis.

- The images are evaluated without user input.
- For this test, single images (not videos) are a prerequisite.
- The test images and the flip_flag.txt file you must use for this test are located in the “DL_Track_US_example/tests/test_images_automatic” folder.

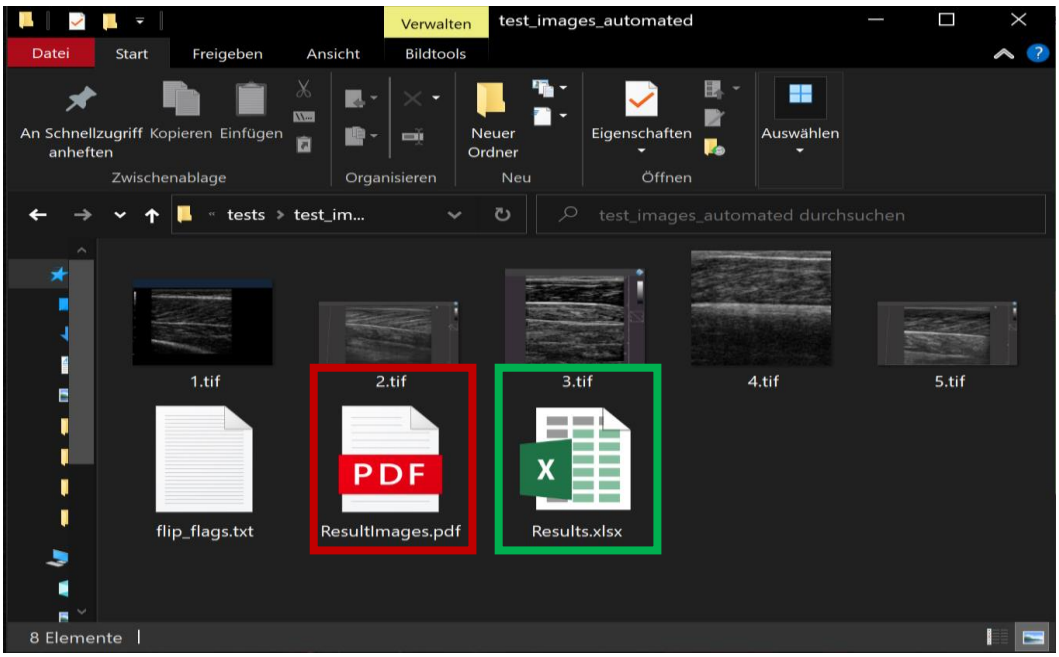
For this test make sure that the files used and parameters specified are exactly as demonstrated below.

- Keep the pre-specified parameter settings in the “Analysis Parameter window” as they are.
- **In v0.2.1 of the GUI, selecht “NO” in the “Filter Fascicles” option.**
- Make sure to use the right **images** (“DL_Track_US_example/tests/test_images_automated”).
- Make sure to use provided pre-trained **models** (“DL_Track_US_example/DLTrack_models”).
- Make sure to use the right **flip_flag.txt file** (“DL_Track_US_example/tests/test_images_automated/flip_flags.txt”).
- Click the **Select parameters** button to set the **analysis parameters** and then the **Run** button to start the analysis.

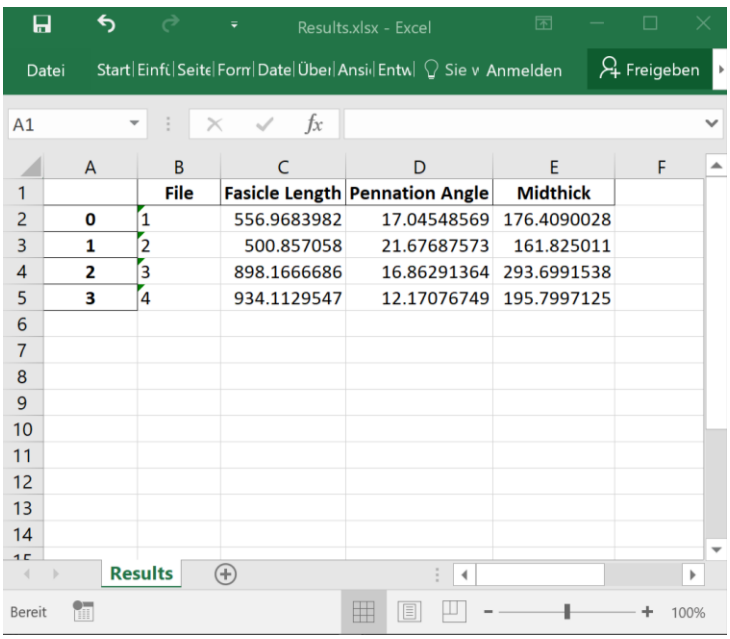


Once the analysis is complete, two new files were created in the "DL_Track_US_example/tests/test_images_automated" folder.

- The **ResultImages.pdf** file.
- The **Results.xlsx** file.




- Open the **Results.xlsx** file and compare the analysis results to the ones demonstrated below.
- If the results are similar, the DL_Track_US package works properly for automated images analysis!




| | A | B | C | D | E | F |
|---|---|---|---|---|---|---|
| 1 | | File | Fasicle Length | Pennation Angle | Midthick | |
| 2 | 0 | 1 | 556.9683982 | 17.04548569 | 176.4090028 | |
| 3 | 1 | 2 | 500.857058 | 21.67687573 | 161.825011 | |
| 4 | 2 | 3 | 898.1666686 | 16.86291364 | 293.6991538 | |
| 5 | 3 | 4 | 934.1129547 | 12.17076749 | 195.7997125 | |

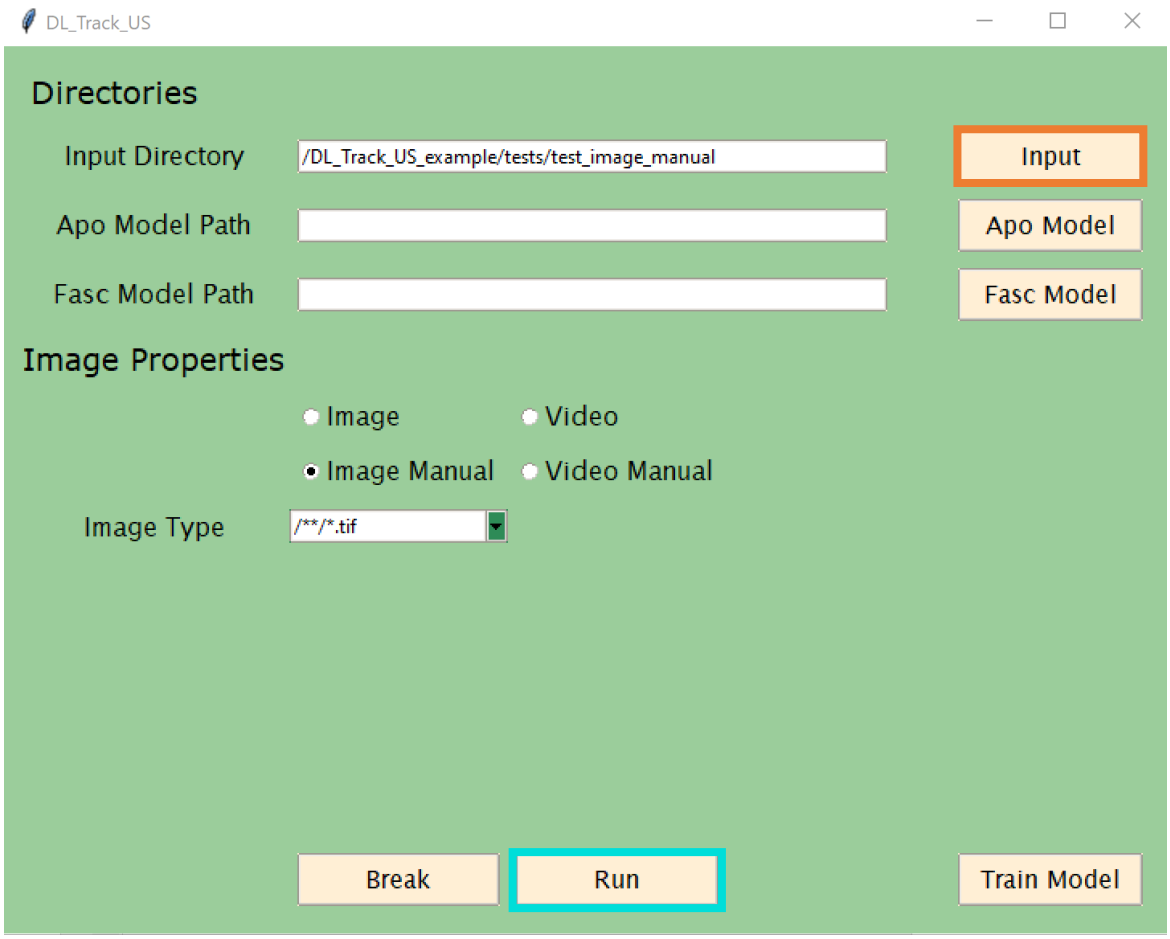
# Manual Image / Video Analysis Test

The next analysis type you are going to test is the manual image / video analysis.

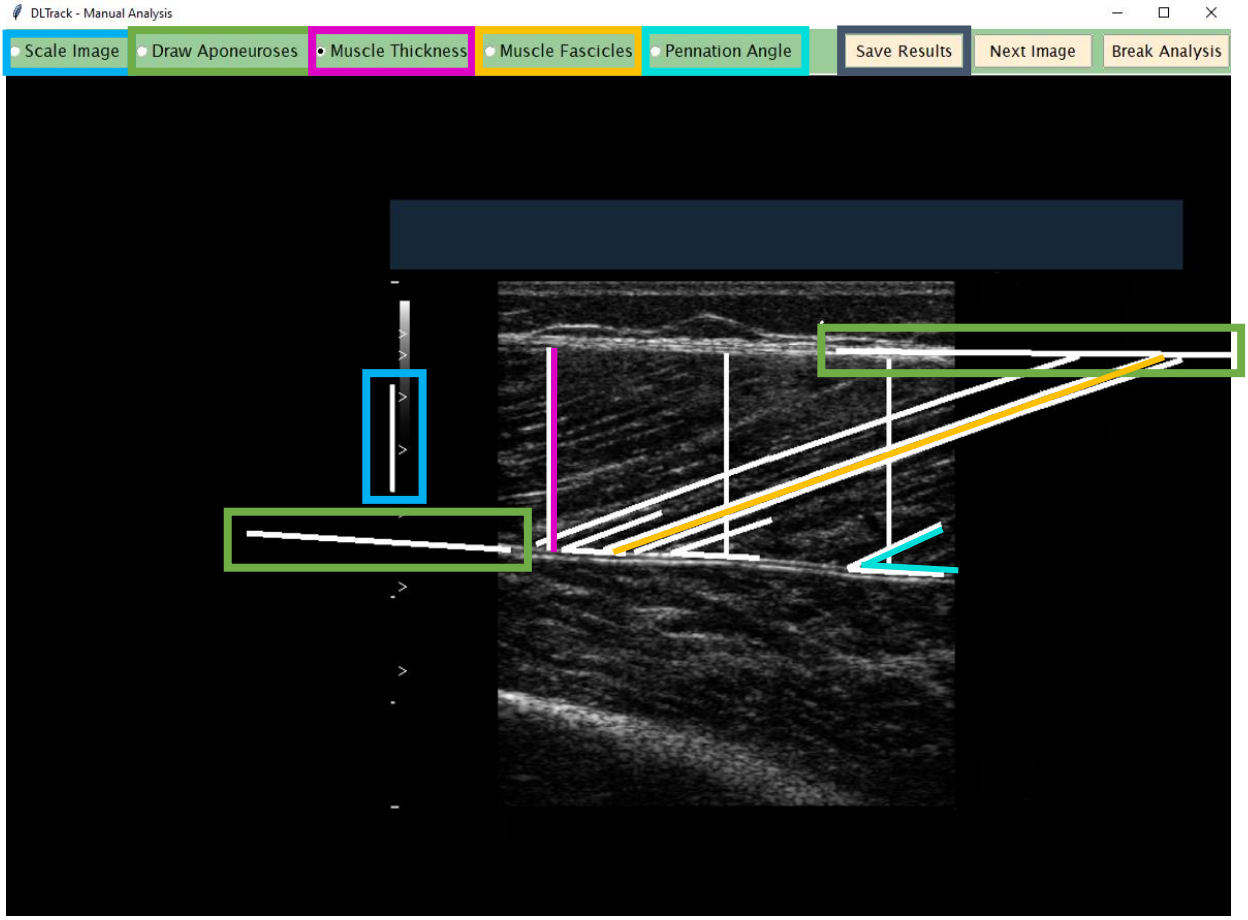
- For this test, single images (not videos) are a prerequisite.
- The test image you must use for this test is located in the "DL_Track_US_example/tests/test_images_manual" folder.
- The analysis types manual image analysis and manual video analysis make use of the same python class (called "ManualAnalysis" and located in the manual_tracing.py file).
- In our opinion, it is thus not necessary to test both analysis types.

For this test make sure that the files used and parameters specified are exactly as demonstrated below.

- Make sure to use the right **image** (“DL_Track_US_example/tests/test_image_manual”).
- Click the **Run** button to start the analysis.



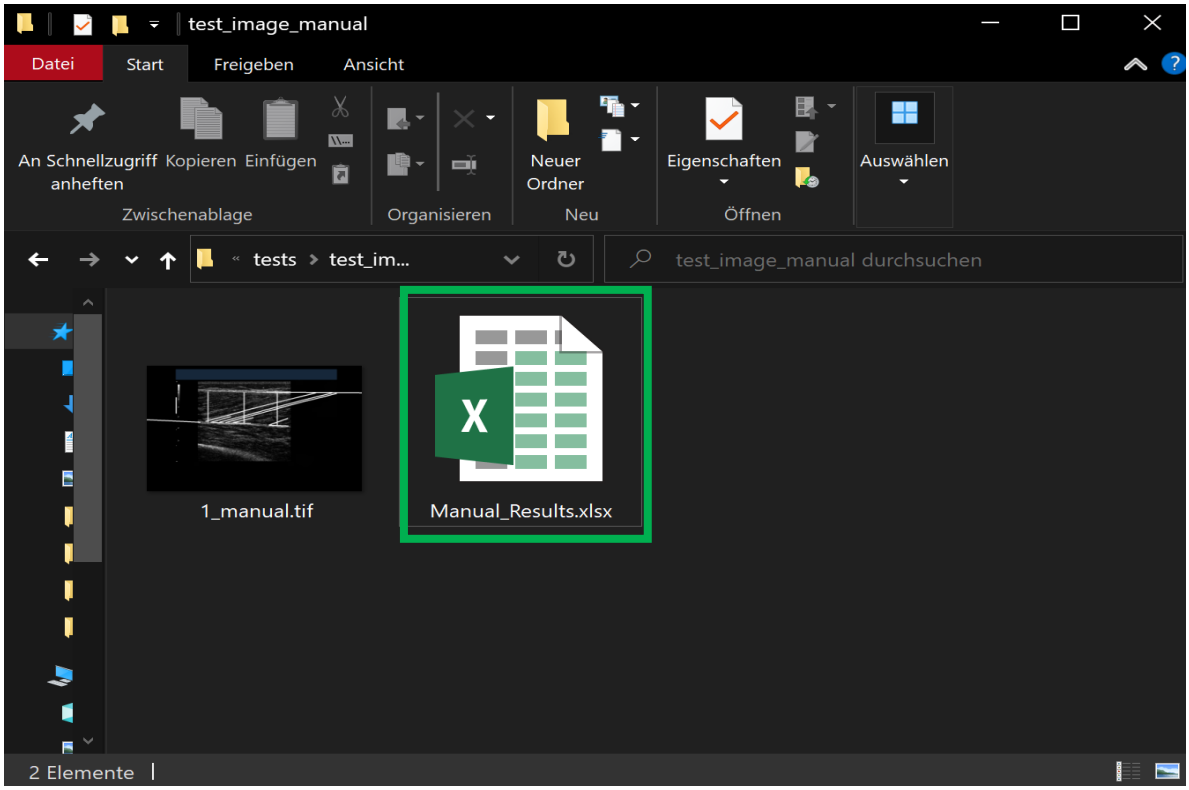
The “Manual Analysis window” should pop up containing the image as demonstrated below.



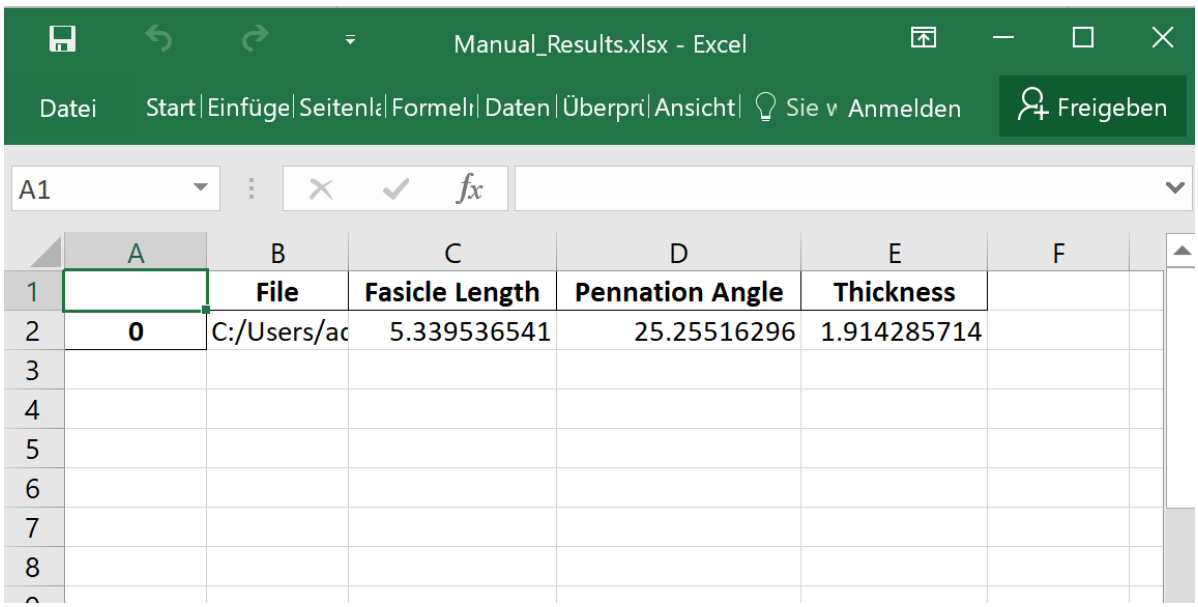

For testing the DL_Track_US manual image / video analysis, simply reanalyse the drawn lines.

- First, **scale the image** by following the one centimetre long **scaling line** in the left of the image.
- Then, **redraw the superficial and deep aponeurosis extension lines**.
- Subsequently, re-analyse the **three vertical muscle thickness lines** using one segment each, the **three diagonal fascicle lines** using three segments each and the **three pennation angles** using two segments each.
- Always choose the Radiobutton corresponding to the parameter you are analysing.
- Once you have re-analysed all the lines image, click on the **Save Results** button to save your analysis results.

One new file was created in the “DL_Track_US_example/tests/test_image_manual”, the **Manual_Results.xlsx** file.




Open the **Manual_Results.xlsx** file and compare the analysis results to the ones demonstrated below. If the results are similar, the DL_Track_US package works properly for manual image / video analysis!




| | A | B | C | D | E | F |
|---|---|---|---|---|---|---|
| 1 | | File | Fasicle Length | Pennation Angle | Thickness | |
| 2 | 0 | C:/Users/ac | 5.339536541 | 25.25516296 | 1.914285714 | |
| 3 | | | | | | |
| 4 | | | | | | |
| 5 | | | | | | |
| 6 | | | | | | |
| 7 | | | | | | |
| 8 | | | | | | |

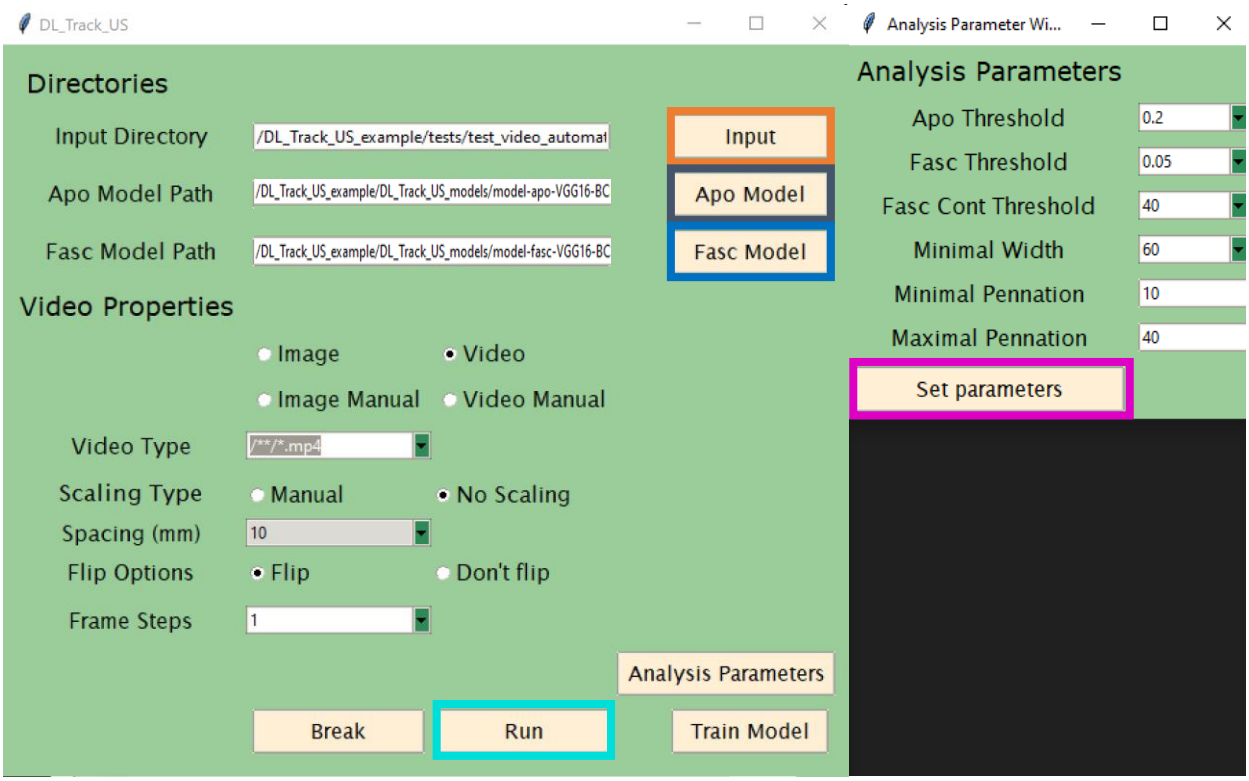
# Automated Video Analysis Test

The next analysis type you are going to test is the automated video analysis.

- Single video frames are evaluated automatically without user input.
- For this test, videos are a prerequisite.
- The test video you must use for this test is located in the "DL_Track_US_example/tests/test_video_automated" folder.

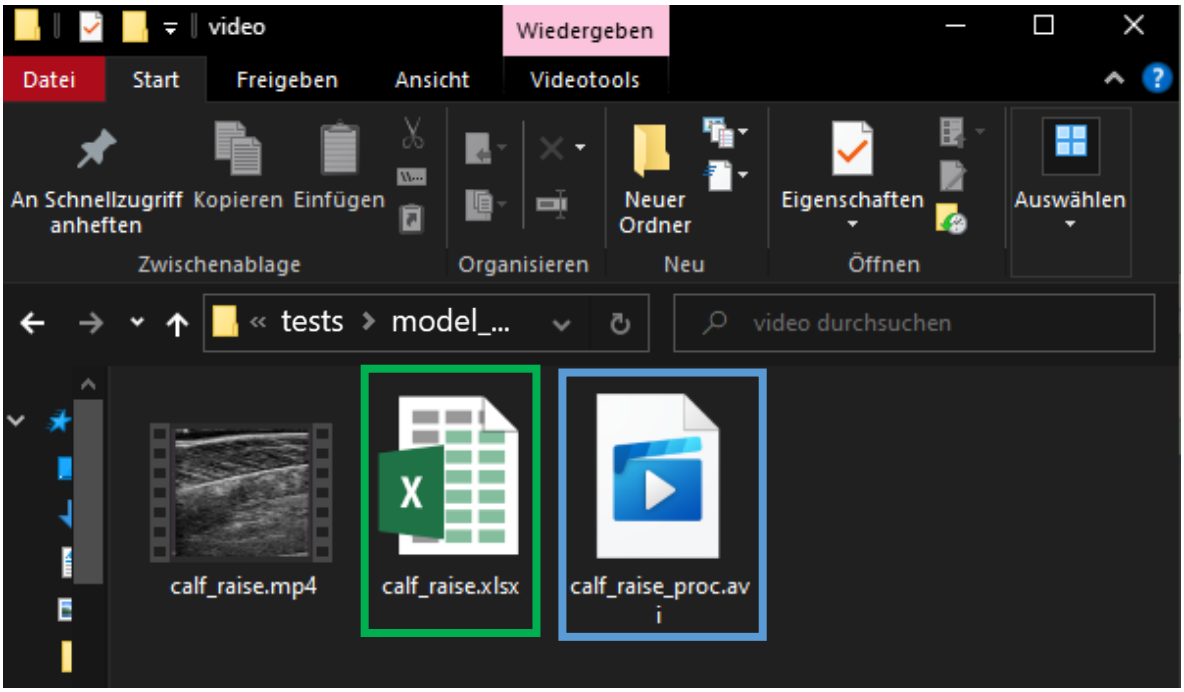
For this test make sure that the files used and parameters specified are exactly as demonstrated below

- Make sure to use the right **video** ("DL_Track_US_example/tests/test_video_automated").
- Make sure to use the provided pre-trained model files (**model-apo-VGG16-BCE-512.h5** & **model-fasc-VGG16-BCE-512.h5**).
- Keep the pre-specified parameter settings in the "Analysis Parameter window" as they are.
- Click the **Select parameters** button to set the analysis parameters.
- Click the **Run** button to start the analysis.

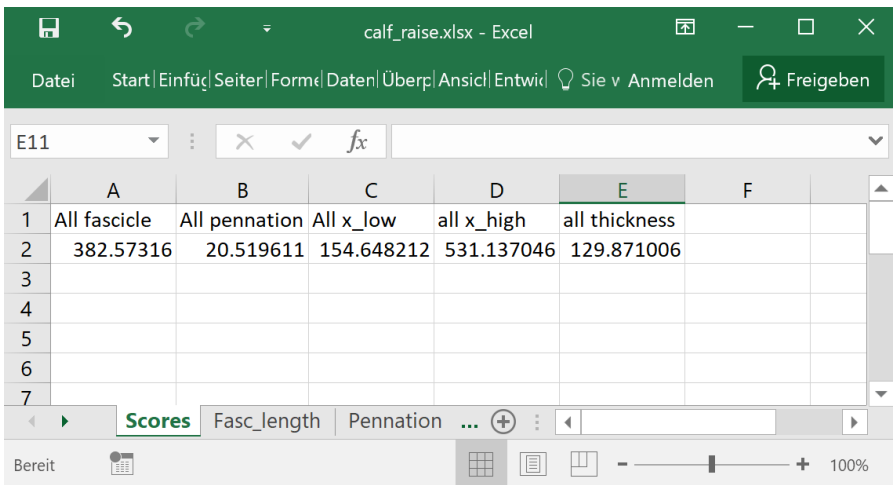


When the analysis is complete, two new files were created in the "DL_Track_US_example/tests/test_video_automated" folder.

- The **calf_raise_proc.avi** file.
- The **calf_raise.xlsx** file.




- Open the **calf_raise.xlsx** file.
- Take the average value from all calculated fascicle length values in all frames, all calculated pennation angles in all frames, all calculated muscle thickness values in all frames and all calculated upper (x_high) and lower (x_low) aponeuroses edge coordinates in all frames.
- If the results are similar to those demonstrated below, the DL_Track_US package works properly for automated images analysis!



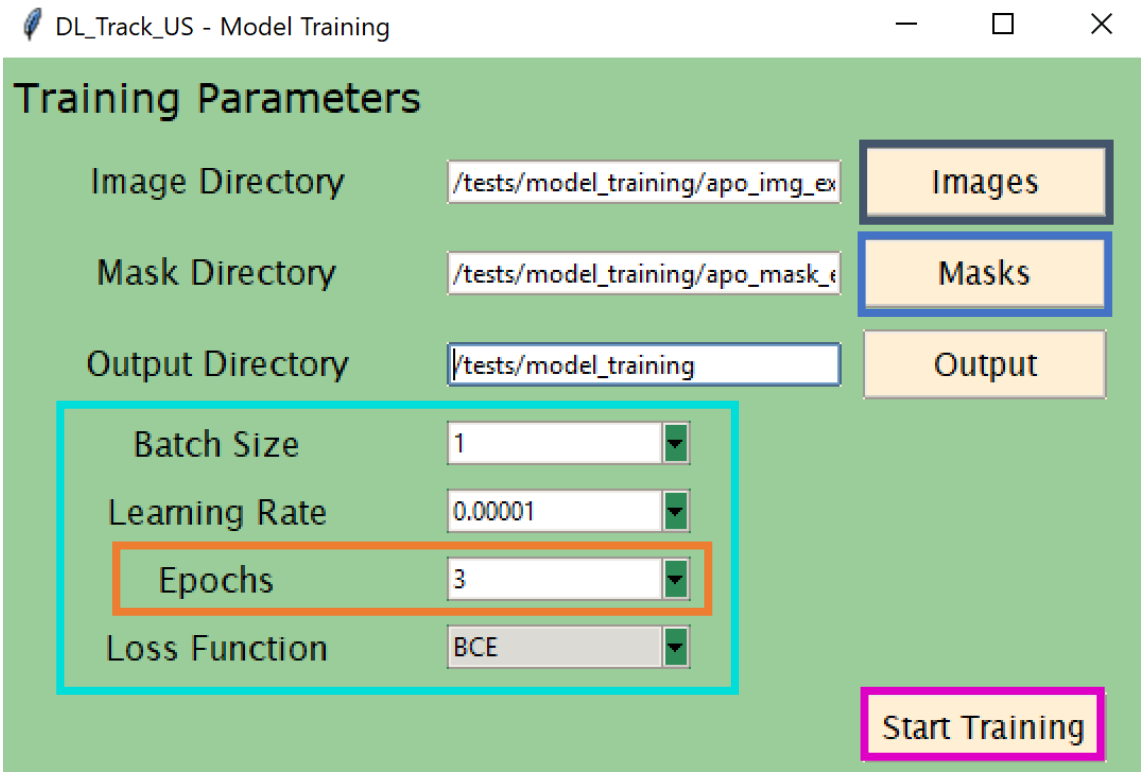
# Model Training Test

In the next few pages, you are going to test the model training using the GUI.

- It is advantageous for model training testing to have a working GPU setup, otherwise model training takes much longer.
- How to setup you GUI for DL_Track_US is described in the installation guidelines of our Github repository.
- The test training images and masks you must use for this test are located at “DL_Track_US_example/tests/model_training” folder.

For this test make sure that the files used and parameters specified are exactly as demonstrated below.

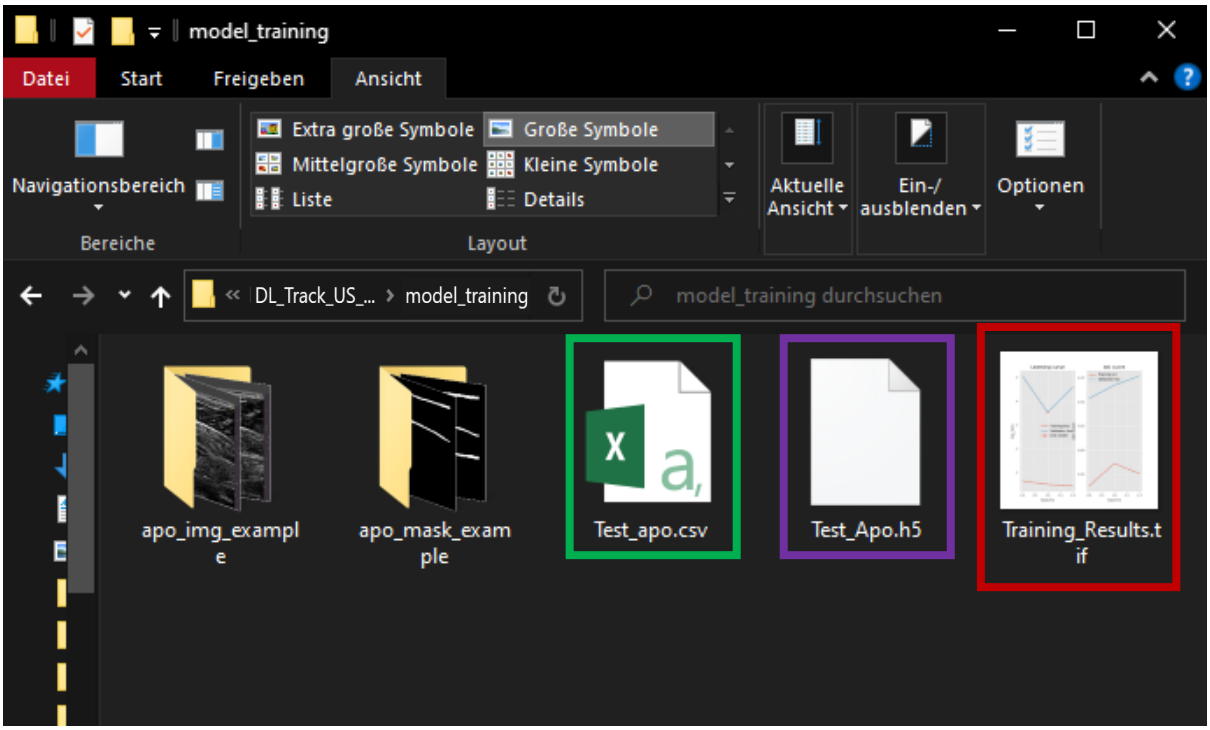
- Click on the “Train Model” button in the main GUI.
- Since you will only make use of the “Model Training window” you can disregard the main GUI.
- Keep the **pre-specified parameter settings** in the “Model Training window” as they are shown below.
- Especially make sure that the number of **Epochs** is 3 (otherwise training for test purposes takes to long).
- made sure to use the right **training images** (“DL_Track_US_example/tests/model_training/apo_img_example”)
- Make sure to use the right **training masks** “DL_Track_US_example/tests/model_training/apo_mask_example”).
- Click the **Start Training** button to start the training process.




Several messageboxes will appear during the training process. Always click “OK”. The messageboxes simply tell you that the images and masks have successfully been loaded, the model was successfully compiled and that the analysis was successfully completed.

When the analysis is complete, three new files were created in the specified output folder.

- The **Test_apo.xlsx** file.
- The **Test_apo.h5** file.
- The **Training_results.tif** file.




Since each training process results in slightly different models, we cannot directly compare your results to ours. However, if the three files were created in the "DL_Track_US_example/tests/model_training" folder, the DL_Track_US package works properly model training!

# Closing remarks

Thanks for checking out the DL_Track_US python package test instructions. We sincerely hope that all tests were successful and you were also able to enjoy it a bit. Moreover, we hope our instructions were clear, concise and easy to follow. In case something was not clearly illustrated at some point, please let us know. Don't hesitate to report this in the Q&A section in the DL_Track_US discussion forum. Otherwise, you can contact us by email at paul.ritsche@unibas.ch.