

# ***DL\_Track***

## ***v0.1.0***

# ***Preface***

Welcome to the DL\_Track python package tutorial. In the next roughly 80 pages, you will learn how to automatically and manually analyse ultrasonography images and videos of human lower limb muscles. You will do so by making extensive use of the graphical user interface provided by the in the DL\_Track package. Moreover, you will learn how to train your own neural networks using the graphical user interface as well. Have fun!

# *Content*

1.	Good to know...	4
2.	Starting the graphical user interface	5
3.	Automated Image Analysis	8
4.	Manual Image Analysis	26
5.	Automated Video Analysis	43
6.	Manual Video Analysis	60
7.	Training your own network	65

# Good to know...

All relevant instructions and guidelines for the installation of the DL\_Track software package are described in our [Github repository](#), so please take a look there if anything is unclear. We have also provided information on what to do when you encounter problems during the installation process, encounter errors during the analysis process that are not caught by the GUI (no error message pop ups and advises you what to do), if you want to contribute to the DL\_Track software package, and how you can reach us.

So, what is DL\_Track all about? The DL\_Track algorithm was first presented by Neil Cronin, Olivier Seynnes and Taija Finni in 2020. The algorithm makes extensive use of fully convolutional neural networks trained on a fair amount of ultrasonography images of the human lower limb. Specifically, the dataset included longitudinal ultrasonography images from the human gastrocnemius medialis, tibialis anterior, soleus and vastus lateralis. The algorithm is able to analyse muscle architectural parameters (muscle thickness, fascicle length and pennation angle) in both, single image files as well as videos. By employing the deep learning models, the DL\_Track algorithm is one of the first fully automated algorithms, requiring no user input during the analysis. Then in 2022, we (Paul Ritsche, Olivier Synnes, Neil Cronin) have updated the code and deep learning models substantially, added a graphical user interface, manual analysis and an extensive documentation. Moreover we turned everything into an openly available Pypi package.

Before we start with this tutorial, here are some important tips:

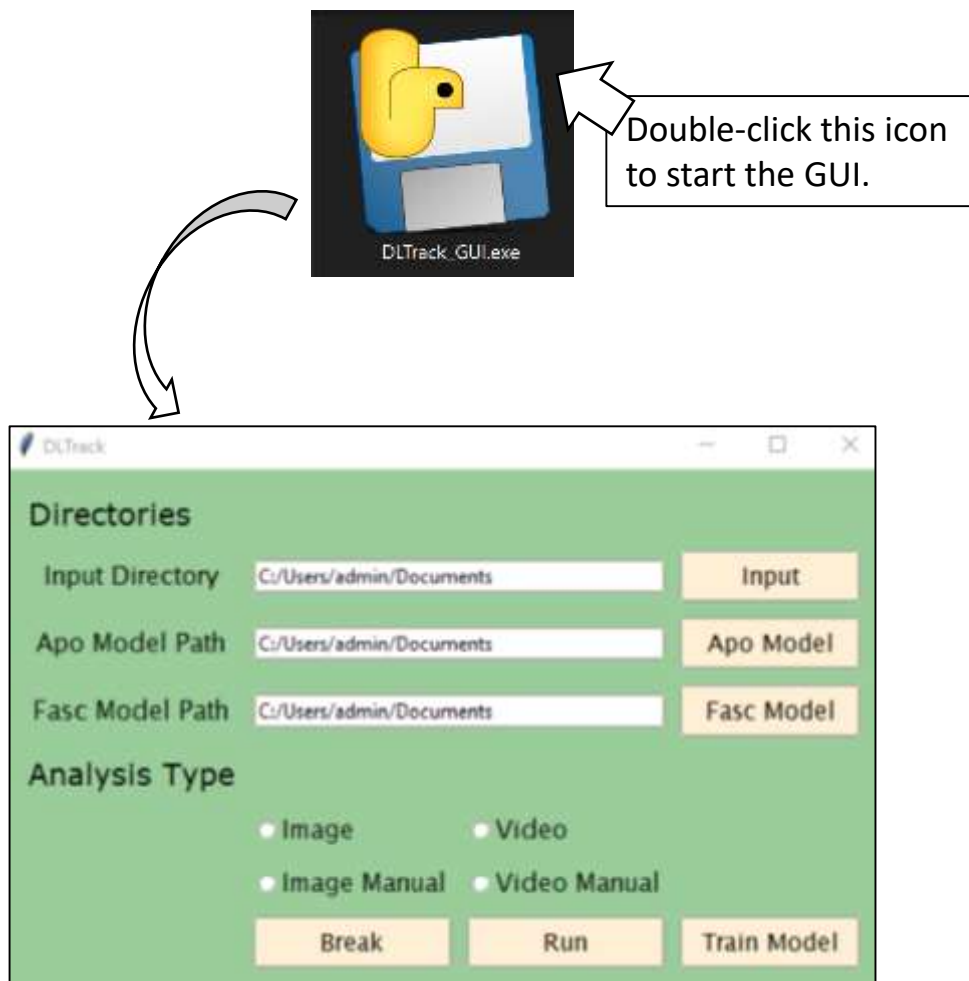
- In case you plan an analysis on images taken from different muscles, we strongly advise to test the algorithm first and in case of bad performance, train your own models. We have provided extensive documentation on how to do so in the “DL\_Track\_training.pdf” file in the DL\_Track/docs (LINK) folder.
- Although we used extensive data augmentation during the model training process, we must caution you about the generalizability of our models. Deep learning is no magic! Even though our model demonstrated good performance on unseen images during testing, we cannot confidently claim that they will work fine on all ultrasonography images. It is possible that even for images of muscles represented in our training data set, different device types, different muscle regions and even different settings of ultrasonography devices during image acquisition might offset model performance.
- Quality matters! Please pay attention that the images you want to analyse with DL\_Track are of high quality. High quality means good image contrast, appropriate image brightness, clearly visible fascicles and aponeurosis and clear alignment of the probe with the fascicle plain. If the quality of the images you want to analyse is bad, the results will be as well.
- Bad model performance can be detected. The first and easiest step to take is to visually inspect the output of the models. If the segmentation results and the actual fascicles and aponeuroses overlap on most of the analysed images, model performance is good. If not, adapt the analysis parameters (how to do so is covered in the tutorials) or train a separate model. Secondly, you should manually analyse a few of your images and compare the model results to your manual results. If both results are similar, model performance is good. If not, adapt the analysis parameters (how to do so is covered in the tutorials) or train a separate model.
- Lastly, we advise to follow the provided testing procedures in the DL\_Track/tests (LINK) folder. This ensures that the DL\_Track package is working properly on you computer.

# Starting the Graphical User Interface

In the very first step of this tutorial, we will take a look at how to start the graphical user interface (GUI) once it was installed. We have provided two different installation procedures: 1. downloading the DLTrack\_GUI.exe file and 2. installing the DL\_Track python package using pip, Github and Pypi.

Let's begin with 1., how to start the GUI when you downloaded the DLTrack\_GUI.exe:

- It doesn't get any easier than this. Navigate to the downloads folder and place the DLTrack\_GUI.exe file somewhere you can easily find it again. Done so, you just have to double click the DLTrack\_GUI.exe file with your left mouse button to start the GUI. Once you've done that, the GUI should open and you are ready to start an analysis.



Now to 2., how to start the GUI when you installed the DL\_Track python package via Pip, Github and Pypi:

There are essentially two ways you can start the GUI. But first lets make sure that the package was correctly installed. The package should be automatically installed when you create the conda virtual environment. Therefore activate the environment by typing “conda activate DL\_Track”:

```
Anaconda Powershell Prompt (miniconda3)
(base) PS C:\Users\admin> conda activate DL_Track_
```

You should see the **activated environment** now in the left round brackets. Next type “conda list” to see all packages installed in the DL\_Track environment:

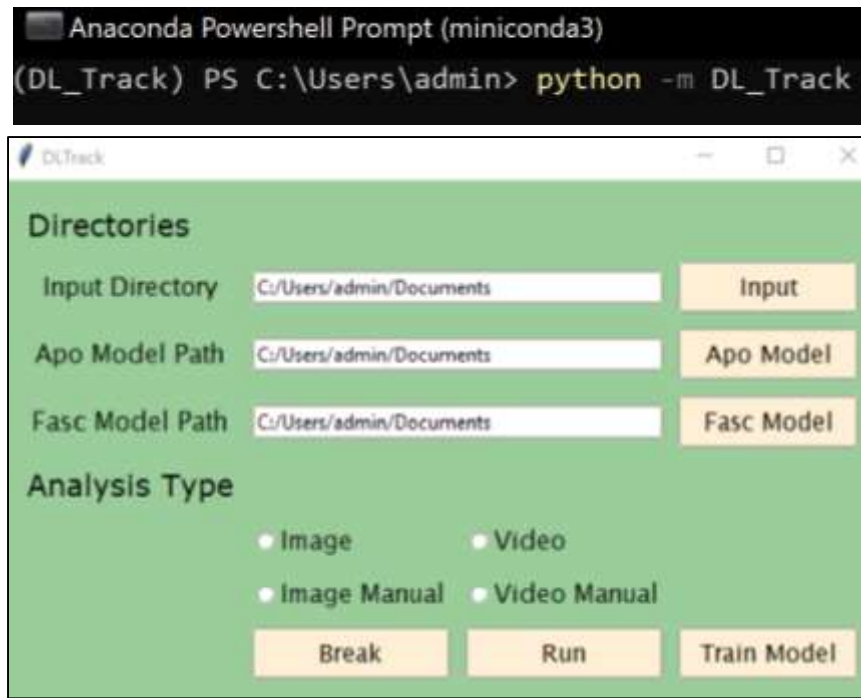
```
Anaconda Powershell Prompt (miniconda3)
(DL_Track) PS C:\Users\admin> conda list_
```

When you pressed enter, all packages installed in the environment should be listed. When **DL\_Track** is included, you are good to go.

```
Anaconda Powershell Prompt (miniconda3)
charset-normalizer 2.1.1 pypi_0 pypi
click 8.1.3 pypi_0 pypi
click-log 0.4.0 pypi_0 pypi
colorama 0.4.6 pypi_0 pypi
commonmark 0.9.1 pypi_0 pypi
contourpy 1.0.5 pypi_0 pypi
cycler 0.11.0 pypi_0 pypi
debugpy 1.6.3 pypi_0 pypi
decorator 5.1.1 pypi_0 pypi
defusedxml 0.7.1 pypi_0 pypi
dl-track 0.1.0 pypi_0 pypi
entrypoints 0.4 pypi_0 pypi
et-xmlfile 1.1.0 pypi_0 pypi
executing 1.1.1 pypi_0 pypi
fastjsonschema 2.16.2 pypi_0 pypi
filelock 3.8.0 pypi_0 pypi
flatbuffers 22.9.24 pypi_0 pypi
fonttools 4.38.0 pypi_0 pypi
future 0.18.2 pypi_0 pypi
gast 0.4.0 pypi_0 pypi
google-auth 2.13.0 pypi_0 pypi
google-auth-oauthlib 0.4.6 pypi_0 pypi
google-pasta 0.2.0 pypi_0 pypi
grpcio 1.50.0 pypi_0 pypi
h5py 3.7.0 pypi_0 pypi
identify 2.5.8 pypi_0 pypi
idna 3.4 pypi_0 pypi
imageio 2.22.2 pypi_0 pypi
```

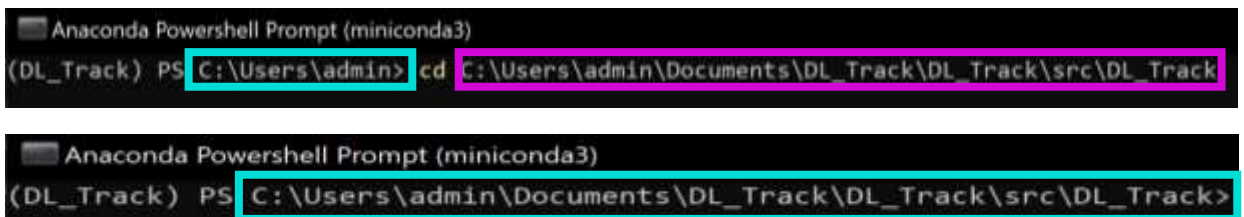
In case the DL\_Track package is not installed in your environment, please take a look at the installation guidelines again. If you still encounter a problem, ask a question in the [Q&A discussion section of DL\\_Track](#) on Github and add the Label “Problem”.

Alright, let's actually start the GUI now once you've made sure that the DL\_Track package is included in your active environment. As mentioned, you can do this in two ways. **First option**, you can run the DL\_Track package from the command prompt. For this, type "python -m DL\_Track" in the command prompt. When you press enter, the GUI should open.



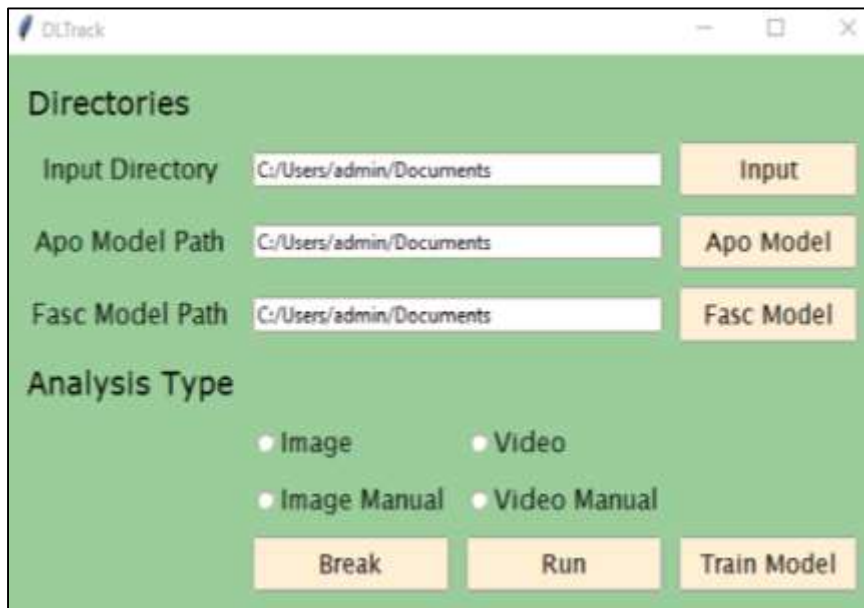
Neither to activate the environment (see previous page) nor to start the package do you need to be in a specific folder. The only prerequisite for this method is an active environment containing all dependencies for as well as the DL\_Track package itself.

**Second option**, you run the python file containing the main GUI manually. Therefore you need all the source files from the Github repository. Please take a look at the [installation guidelines](#) on how to get them. It is required that the environment containing all dependencies for as well as the DL\_Track package itself is active (see previous page) and that you are in the folder containing the DLTrack\_GUI.py module. To navigate to the folder type "cd **yourpathtofile**" in the command prompt. By pressing enter, the **path** is changed to the folder containing the DLTrack\_GUI.py file. Note that the end of the path "/DL\_Track/src/DL\_Track" given in the example above should be the end of your specified path as well.



To start the GUI, type “python DLTrack\_GUI.py” in the command prompt once you have activate the environment containing the DL\_Track package and all its dependencies and navigated to the folder that contains the DLTrack\_GUI.py file. By pressing enter, the GUI should start.

```
Anaconda Powershell Prompt (miniconda3)
(DL_Track) PS C:\Users\admin\Documents\DL_Track\DL_Track\src\DL_Track> python DLTrack_GUI.py
```



Congrats, you now know all the ways how to start the DL\_Track GUI! Which way you use to start the GUI might be dependent on your usage intentions. If you just want to use the GUI and analyse images / videos, it is completely enough to download the DLTrack\_GUI.exe and start right away. This is definitely the easiest way to start the GUI. However, if you want to customize the code and maybe even contribute to further releases, you are required to download all the source files in order to be able to change anything. To directly see your implemented changes use the last option explained below to start the GUI.

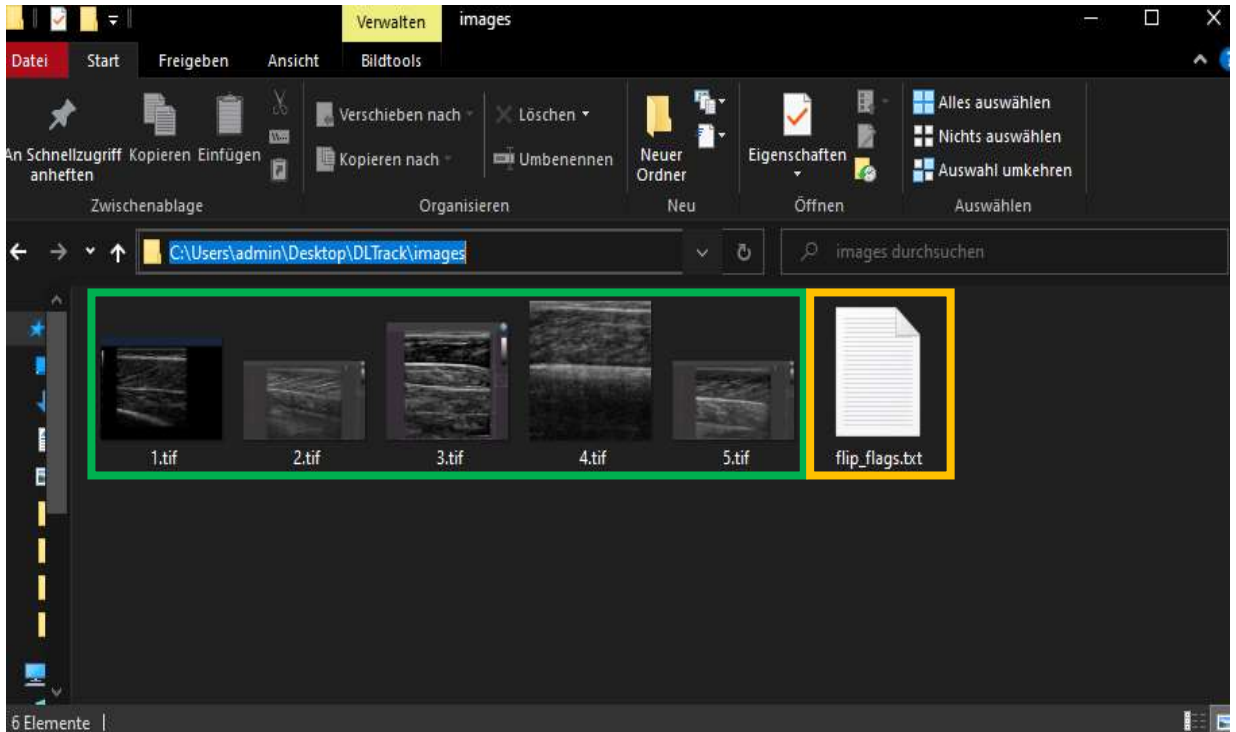


# ***Automated Image Analysis***

The DL\_Track python software package offers several different analysis types for analysis of human lower limb longitudinal ultrasonography images. The first analysis type this tutorial covers is the automated image analysis. The images are evaluated without user input and may be scaled. Scaling the images will ensure estimated muscle architectural parameters are converted to centimetre units. For this type of analysis, single images (not videos) are a prerequisite. These images should be contained in a single folder, like in the “DL\_Track\_examples/images” folder. If you haven’t downloaded this folder, please do so now (link: [.](#) ). Unzip the folder and put it somewhere accessible, for example on your desktop. We will make use of the included example files extensively during this tutorial. Moreover, a FlipFlag.txt file is needed. Such a file is included in the “DL\_Track\_examples/images” folder as well. In the next few pages, we will look at every required step to successfully perform automated image analysis with DL\_Track.

# 1. Creating image directory & FlipFlag.txt file

In order for DL\_Track to recognize your images, they should best be in a single folder (though one subfolder structure is acceptable as well). Take a look how you might structure this:



You can see in the picture above that the folder contains **5 images**, a **flip\_flag.txt** file and is **located on the desktop**. This structure is already included in the “DL\_Track\_example” folder. It is not required to have the **flip\_flag.txt** file in the same folder as the **images** to be analysed, but it is convenient.

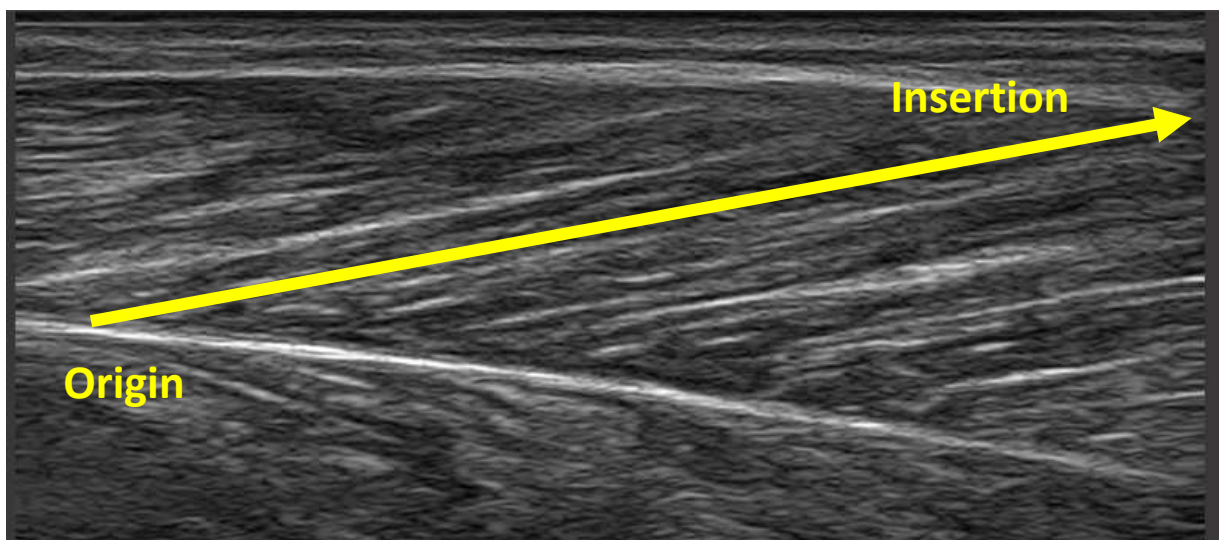
Lets take a closer look at the **flip\_flags.txt** file and how this should look like in order for the analysis to work. Below you can see an the **flip\_flag.txt** file in the directory.



In the flip\_flag.txt file you can see the respective **flip-flag** for each image. For every image there must be a **flip-flag**. If the number of **flip-flags** and **images** doesn't match, an error is raised. Another possible way to specify is displayed below. This is relevant when multiple subfolders are included, as each line then represents a subfolder.



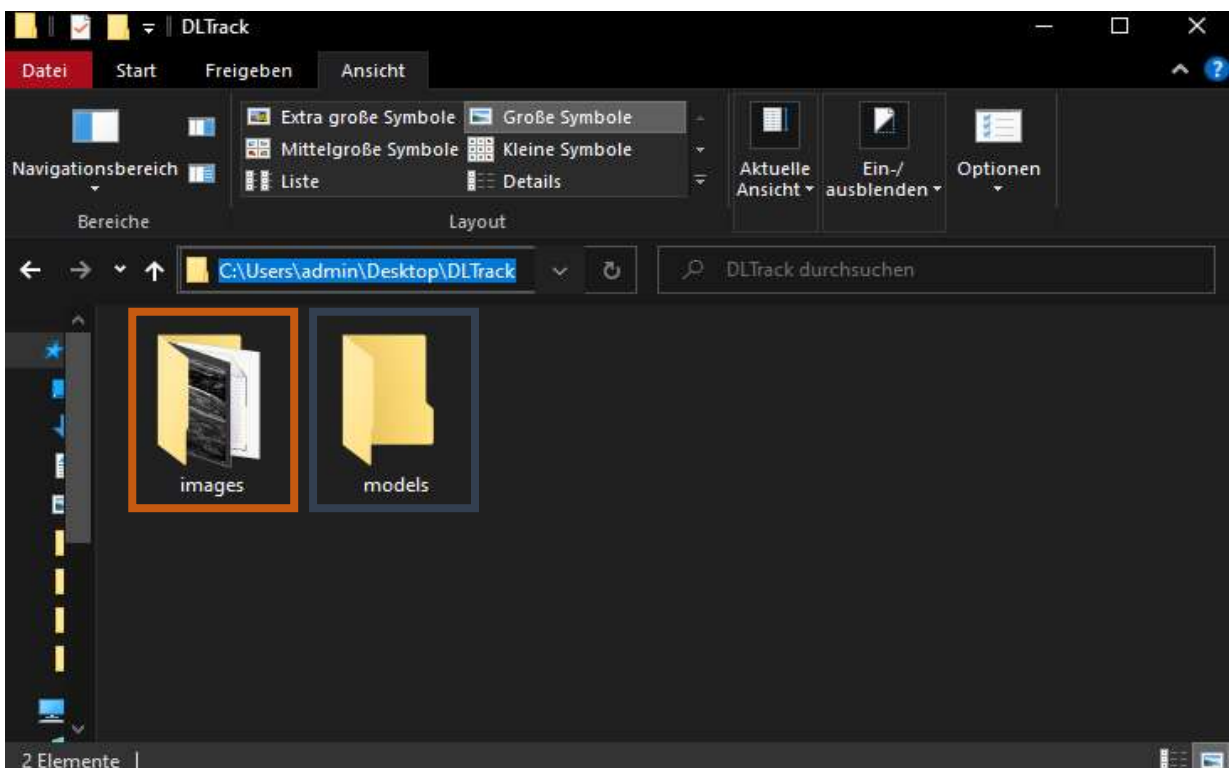
The **flip-flag** determines if an **image** is flipped during analysis or not. A "0" stands for no flipping, whereas "1" means please flip the image. The **flip-flags** enable that each **image** can be automatically flipped or not. None of the example **images** must be flipped. Their fascicle orientation is correct, with fascicles originating at the bottom left and inserting on the top right. Here is a visual representation of a **correct** fascicle orientation. If the fascicles in your **images** are orientated differently, please specify a "1" as a **flip-flag** for those **images**.



Once you have created a directory containing a flip\_flag.txt file with a **flip-flag** for each **image** in the directory that “0” for **images** with correct orientation and “1” for incorrectly orientated **images**, we can get to the analysis part. (You actually do not need to anything at this point, because you can use the example **images** folder “DL\_Track\_examples/images” with it’s contained images and flip\_flag.txt file.)

## 2. Creating Neural Network Directories

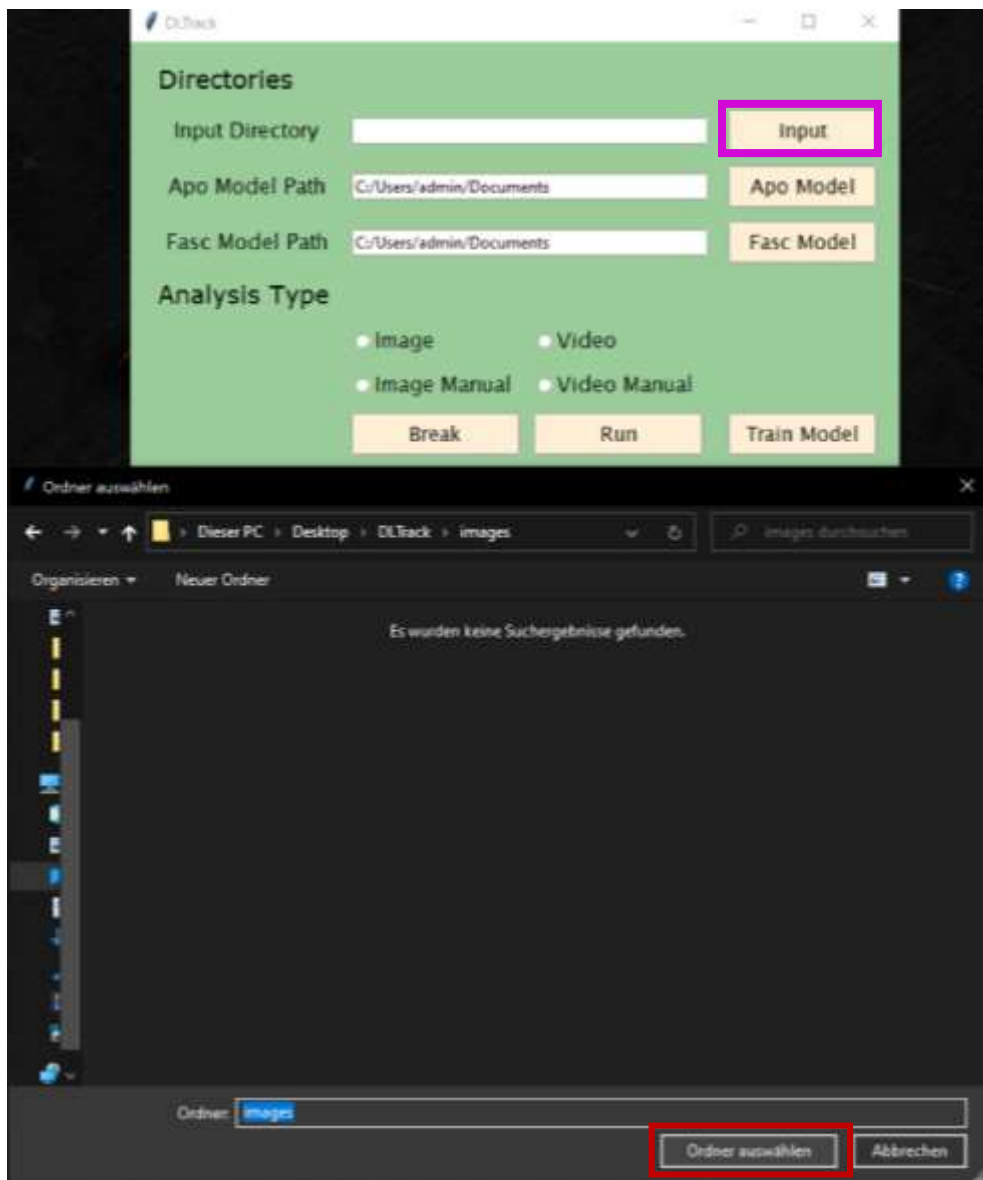
Before we start with the analysis, we need to create a directory for the pre-trained neural networks. The **folder containing the images** (in this case the “DL\_Track\_examples/images” folder) is already included in the “DL\_Track\_example” folder. We will now create a separate folder for the pre-trained **aponeurosis and fascicle neural networks**. In case you have not downloaded the models, please do so now (link: ). Place them in a subfolder of the “DL\_Track\_example” folder for the moment, like “DL\_Track\_example/**models**”. You will make use of these neural networks later as well, when you analyse your own images outside of this tutorial. Of course you can move them to a different folder then. Your folder structure inside the “DL\_Track\_example” folder should now look something like this with the neural networks placed in the **models** folder.



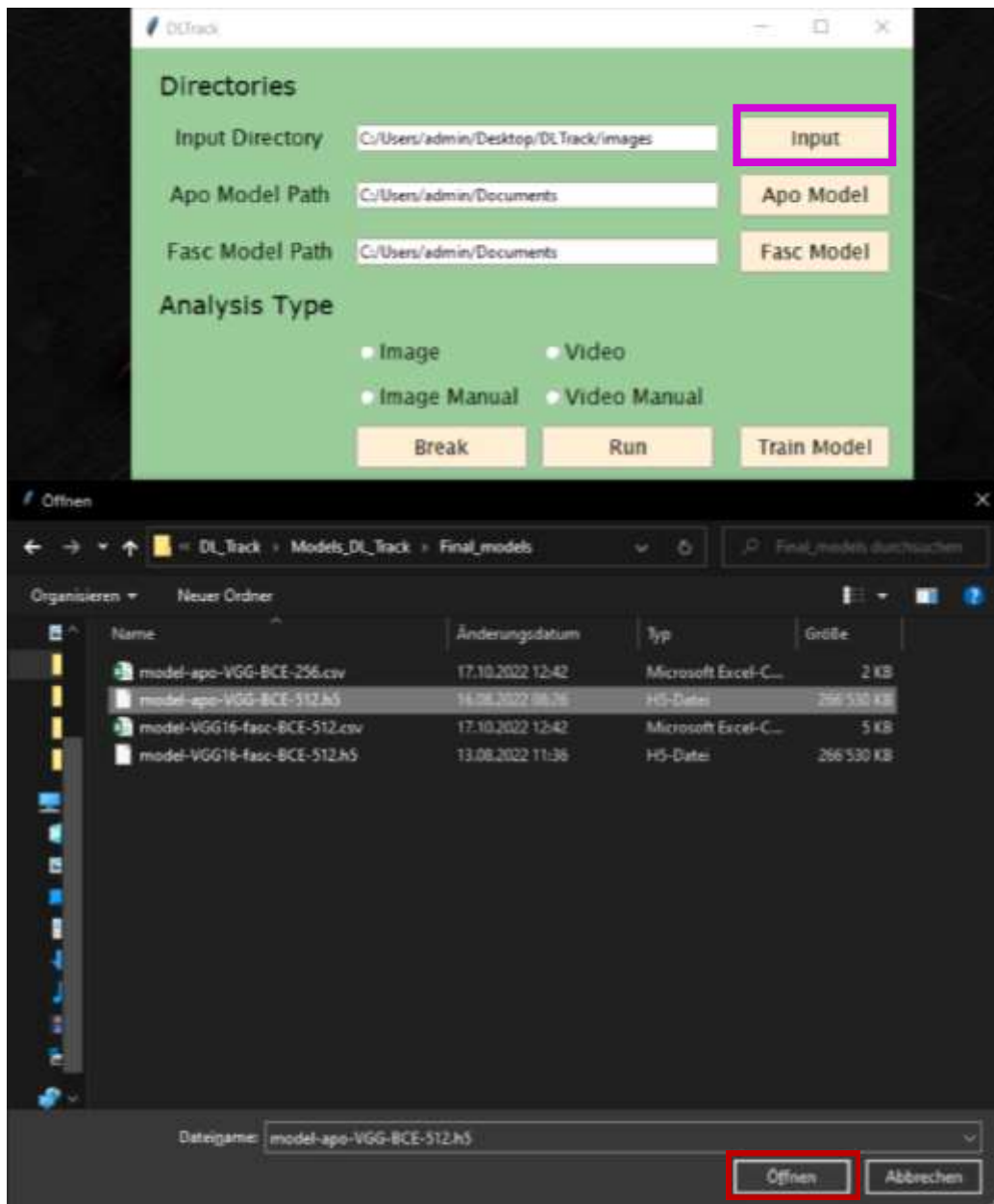
### 3. Specifying Input Directories in the GUI

Finally you can start with the actual analysis! The first step of every analysis type in DL\_Track is to specify the input directories in the graphical user interface (GUI). We assume that you have already opened the GUI. (If not, take a look at the previous chapter of this document “Starting the GUI”).

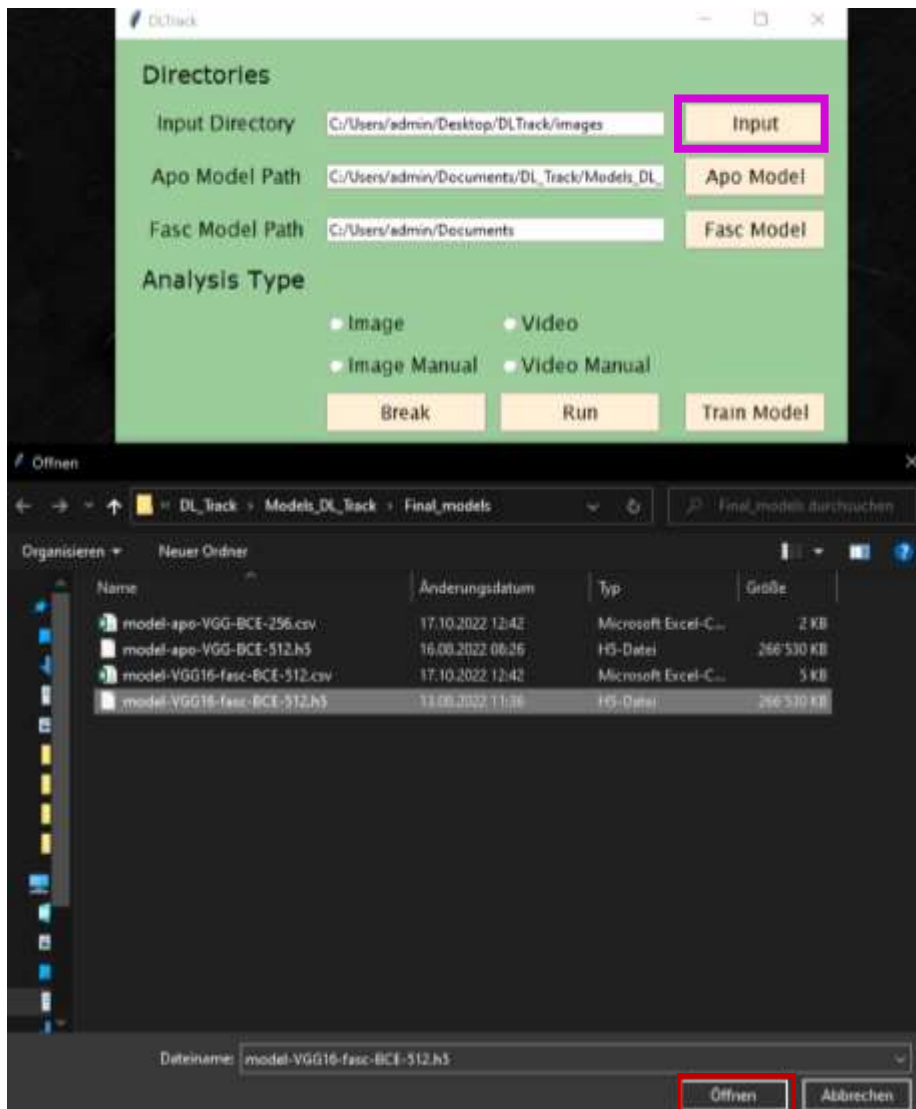
You will begin with specifying the path to the folder containing the **images** to be analysed. Remember this was the folder “DL\_Track\_example/**images**”. By clicking on the **Input** button in the GUI a selection window opens where you need to select the images folder. Click **select folder** to specify the path in the GUI.



Next, you will specify the absolute path to the **aponeurosis neural network**. Remember that you placed it in the “DL\_Track\_example/models”. By clicking on the **Apo Model** button in the GUI a selection window opens where you need to select the **aponeurosis neural network** in the models folder. Click **open** to specify the path to the **aponeurosis neural network** in the GUI.



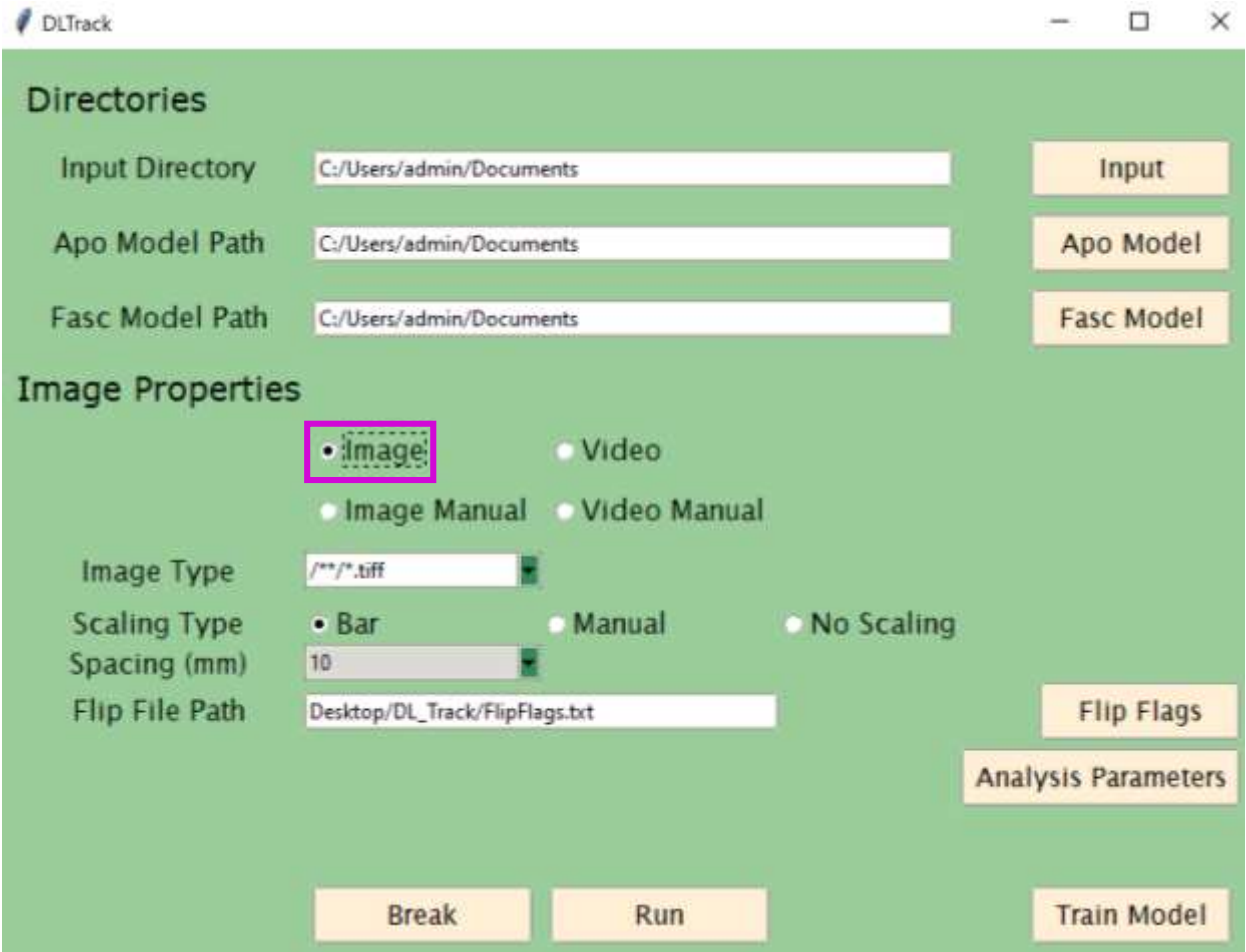
Next, you will specify the absolute path to the **fascicle neural network**. Remember that you placed it in the “DL\_Track\_example/**models**”. By clicking on the **Fasc Model** button in the GUI a selection window opens where you need to select the **fascicle neural network** in the models folder. Click **open** to specify the path to the **fascicle neural network** in the GUI.



You have now successfully defined all the input directories required for automated image analyses with DL\_Track. In the next section you will specify all relevant analysis parameters, including the analysis type. We will also explain what each parameter is used for.

## 4. Specifying Relevant Parameters

As a first step, you will select the right analysis type in the GUI. Since this section is about automated image analysis, please select the **Image** radiobutton. You can see that the GUI unfolds and several other parameters appear. You will set those in the next steps on the next page.



The screenshot shows the DLTrack GUI with a green background. The 'Directories' section has three text boxes for 'Input Directory', 'Apo Model Path', and 'Fasc Model Path', all containing 'C:/Users/admin/Documents'. To the right of each box is a button: 'Input', 'Apo Model', and 'Fasc Model'. The 'Image Properties' section has a radio button group with 'Image' selected (highlighted by a red box), 'Video', 'Image Manual', and 'Video Manual'. Below this is a dropdown for 'Image Type' showing '/\*\*/\*.tiff'. There are three radio buttons for 'Scaling Type': 'Bar' (selected), 'Manual', and 'No Scaling'. A 'Spacing (mm)' slider is set to 10. A 'Flip File Path' text box contains 'Desktop/DL\_Track/FlipFlags.txt', with a 'Flip Flags' button to its right. At the bottom are three buttons: 'Break', 'Run', and 'Train Model'. A label 'Analysis Parameters' is positioned above the 'Train Model' button.

DLTrack

**Directories**

Input Directory: C:/Users/admin/Documents **Input**

Apo Model Path: C:/Users/admin/Documents **Apo Model**

Fasc Model Path: C:/Users/admin/Documents **Fasc Model**

**Image Properties**

☒ Image ☐ Video

☐ Image Manual ☐ Video Manual

Image Type: /\*\*/\*.tiff

Scaling Type: ☒ Bar ☐ Manual ☐ No Scaling

Spacing (mm): 10

Flip File Path: Desktop/DL\_Track/FlipFlags.txt **Flip Flags**

**Analysis Parameters**

**Break** **Run** **Train Model**



Next, you need to specify the Image Type. The ending of the **Image Type** must match the ending of your **images**, otherwise no files are found by DL\_Track. You can either select a pre-specified ending from the dropdown list or type your own ending. Please keep the formatting similar to those **Image Types** provided in the dropdown list. All the **images** in the “DL\_Track\_example/**images**” folder are of the **Image Type** “.tif”. Thus, you should select the “/\*\*/\*.\*.tif” **Image Type**.

The image shows the DLTrack software interface with the following configuration options:

- Directories**
  - Input Directory: C:/Users/admin/Documents (Input button)
  - Apo Model Path: C:/Users/admin/Documents (Apo Model button)
  - Fasc Model Path: C:/Users/admin/Documents (Fasc Model button)
- Image Properties**
  - Image Type: /\*\*/\*.\*.tif (highlighted with a red box)
  - Scaling Type: Bar (selected), Manual, No Scaling
  - Spacing (mm): 10
  - Flip File Path: Desktop/DL\_Track/FlipFlags.txt (Flip Flags button)
- Buttons**
  - Analysis Parameters button
  - Break button
  - Run button
  - Train Model button

Subsequently, you need to specify the image **Scaling Type**. Scaling in general has the huge advantage that the resulting estimated muscle architectural features are in centimetre units rather than pixel units. There are three **Scaling Types** in the DL\_Track package. For this tutorial however, you will select the “**No Scaling**” option as displayed below. We will explain the other two **Scaling Types** in the next pages.

The image shows the DLTrack software interface. It has a green background and a white title bar with the text 'DLTrack'. The interface is divided into two main sections: 'Directories' and 'Image Properties'.

**Directories**

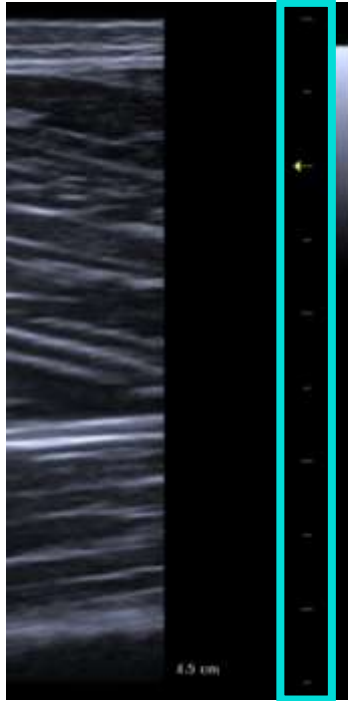
- Input Directory:** C:/Users/admin/Documents (with an 'Input' button to the right)
- Apo Model Path:** C:/Users/admin/Documents (with an 'Apo Model' button to the right)
- Fasc Model Path:** C:/Users/admin/Documents (with a 'Fasc Model' button to the right)

**Image Properties**

- Image Type:** A dropdown menu showing '/\*\*/\*.tif'.
- Scaling Type:** Three radio buttons are present: 'Bar', 'Manual', and 'No Scaling'. The 'No Scaling' option is selected and highlighted with a red rectangle.
- Spacing (mm):** A text input field with the value '10'.
- Flip File Path:** Desktop/DL\_Track/FlipFlags.txt (with a 'Flip Flags' button to the right)

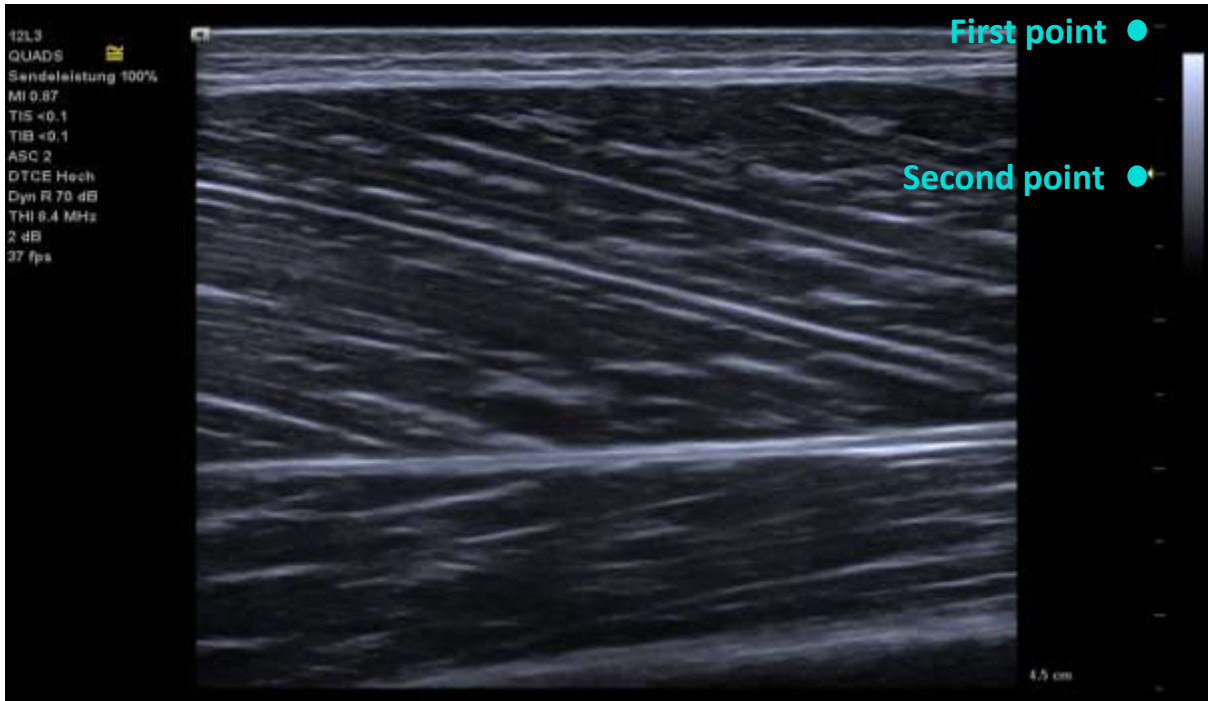
At the bottom of the interface, there are three buttons: 'Break', 'Run', and 'Train Model'. There is also an 'Analysis Parameters' button located to the right of the 'Flip File Path' field.

Another **Scaling Type** except “No Scaling” is “Bar”. This **Scaling Type** is only applicable if there are **scaling bars** in the right side of the ultrasonography image:



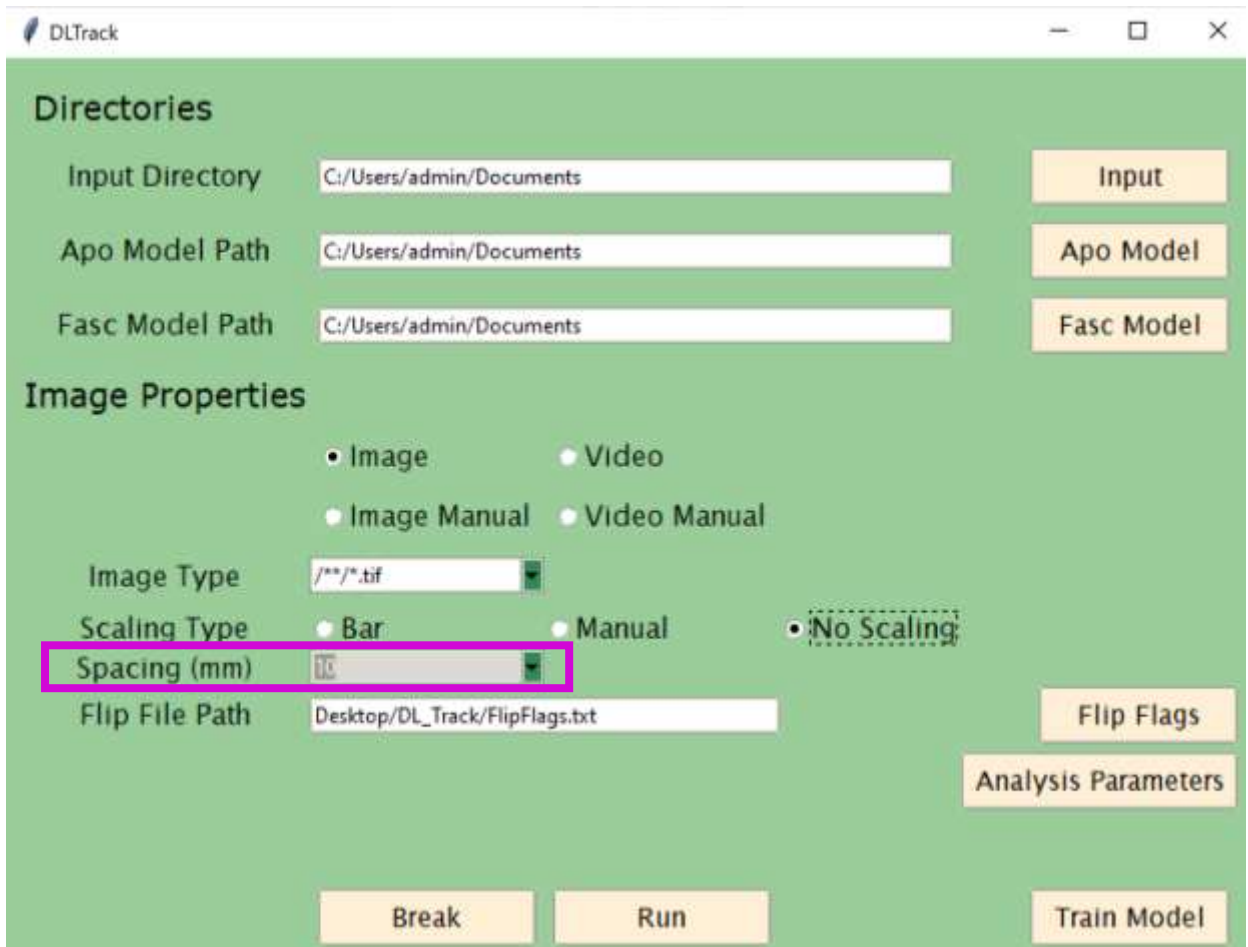
It is not important if the **scaling bars** look exactly like the ones in the above image. They just need to be **next to the image** and **clearly separated** from each other. We advise you to try this **Scaling type** on a few of your images and find out for yourself if it works. Files that cannot be analysed with this **Scaling type** will be recorded in an `failed_images.txt` file in the image input folder.

The last of the three **Scaling Types** is “Manual”. This **Scaling Type** requires input from the user. When you choose “Manual” as your Scaling type, you need to manually place **two points** on the image using the left mouse button.



No worries, you do not actually need to draw on the image. Just click one time with your left mouse button to record the **first point** (nothing will be displayed on the images during actual analysis). Place the **second point** at a known distance of either 5, 10, 15 or 20 millimetre. The distance you chose must be represented in the Scaling (see next page) parameter in the GUI.

Whenever you use **“Bar”** or **“Manual”** as your Scaling Type, please make sure that the minimum distance between the scaling bars or the known distance between you manually specified points is represented in the **Spacing** parameter. For the **“No Scaling”** Scaling type, this is not necessary.

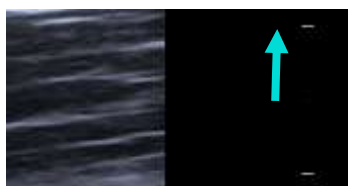


The image shows the DLTrack software interface. It has a green background and a white title bar with the text "DLTrack". The interface is divided into several sections:

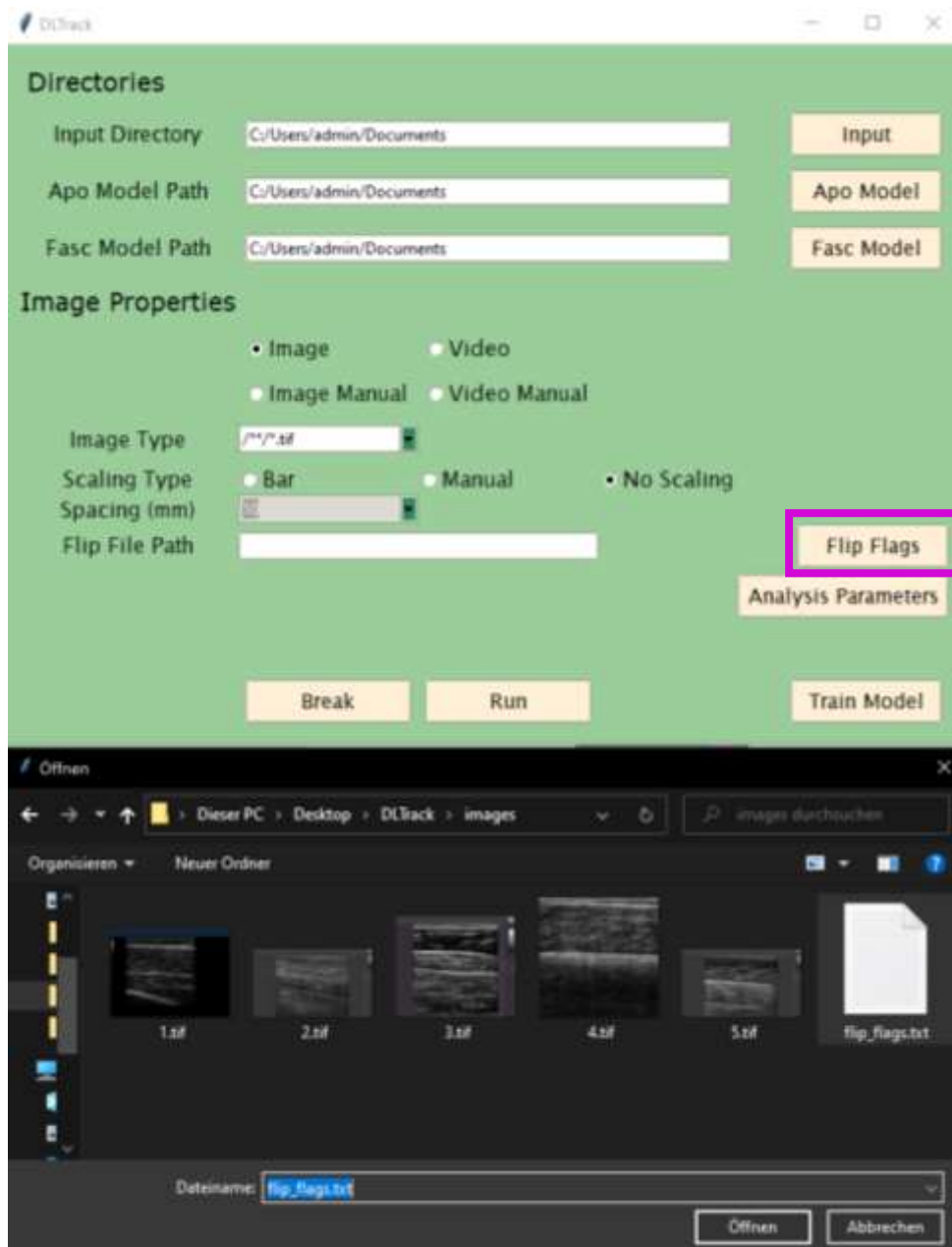
- Directories:** Contains three input fields, all with the path "C:/Users/admin/Documents":
  - Input Directory (with an "Input" button)
  - Apo Model Path (with an "Apo Model" button)
  - Fasc Model Path (with a "Fasc Model" button)
- Image Properties:** Contains radio buttons for "Image", "Video", "Image Manual", and "Video Manual". Below these is an "Image Type" dropdown menu set to "/\*/\*.tif".
- Scaling Type:** Contains radio buttons for "Bar", "Manual", and "No Scaling". The "No Scaling" option is selected and highlighted with a dashed box.
- Spacing (mm):** A dropdown menu is highlighted with a red rectangle, showing a list of values (5, 10, 15, 20).
- Flip File Path:** An input field with the path "Desktop/DL\_Track/FlipFlags.txt" and a "Flip Flags" button.
- Analysis Parameters:** A button.
- Buttons:** At the bottom, there are three buttons: "Break", "Run", and "Train Model".

You can select the **Spacing** parameter only from the dropdown list as 5, 10, 15 or 20 millimetre. For this tutorial it is not necessary to select anything, as the Spacing parameter is not used during an analysis with Scaling Type **“No Scaling”**.

So far, we haven’t explained how to determine the minimal **distance** between the scaling bars in an image. This is simply the **distance** in millimeter between the two nearest scaling bars in the image. If you do not know this **distance**, please use **“Manual”** or **“No Scaling”** Scaling Type. For example in the image from before, the **distance** between the nearest bars is 5 millimetre. We know that because the distance between the bigger bars is always 10 millimetre.

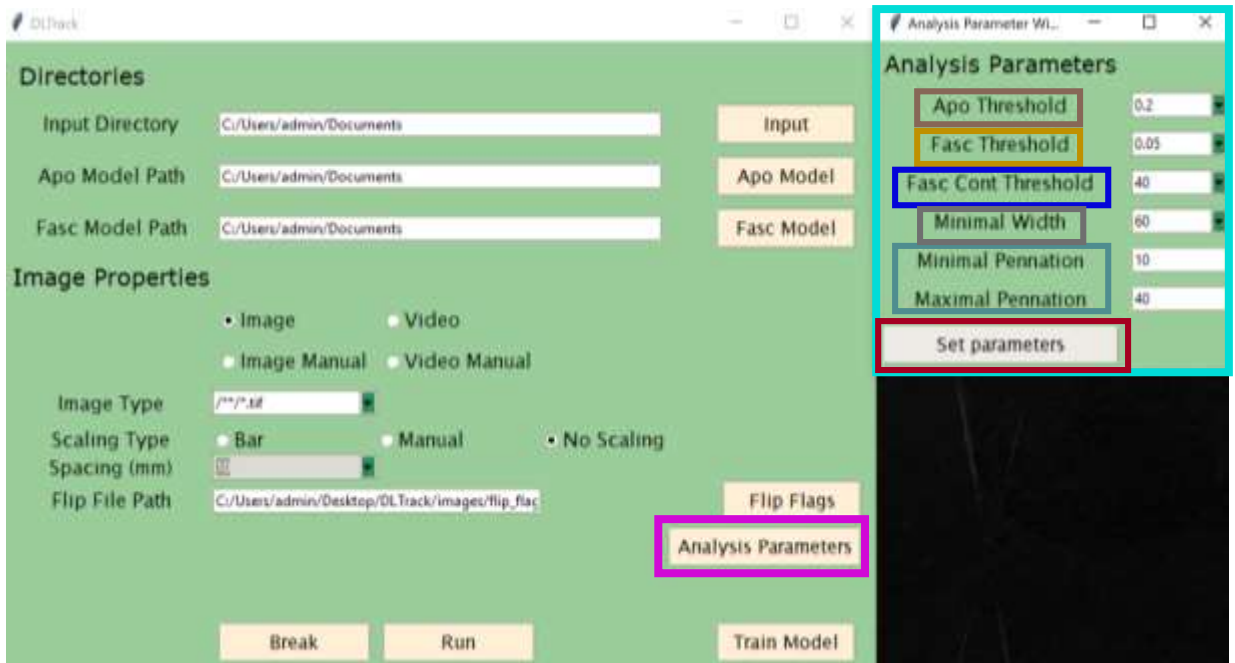


The next in the image analysis using the DL\_Track package is to specify the absolute path to the **flip\_flag.txt** file. This is actually the same procedure than selecting the absolute to the **aponeurosis and fascicle neural networks** before. By clicking the **Flip Flags** button, a dialogue will pop up and you can select the **flip\_flag.txt** file to retrieve it's path. In this example, the **flip\_flag.txt** file is located at "DL\_Track\_example/images". Remember, the amount of **flip-flags** in the flip\_flag.txt file must equal the amount of **images** in the **images** folder.



## 5. Specifying Analysis Parameters

Awesome, you have now successfully selected all relevant parameters in the main GUI window. As a LAST step, you need to specify the analysis parameters for the **aponeurosis and fascicle neural networks**. When you press the **Analysis Parameters** button, a separate window will pop-up.



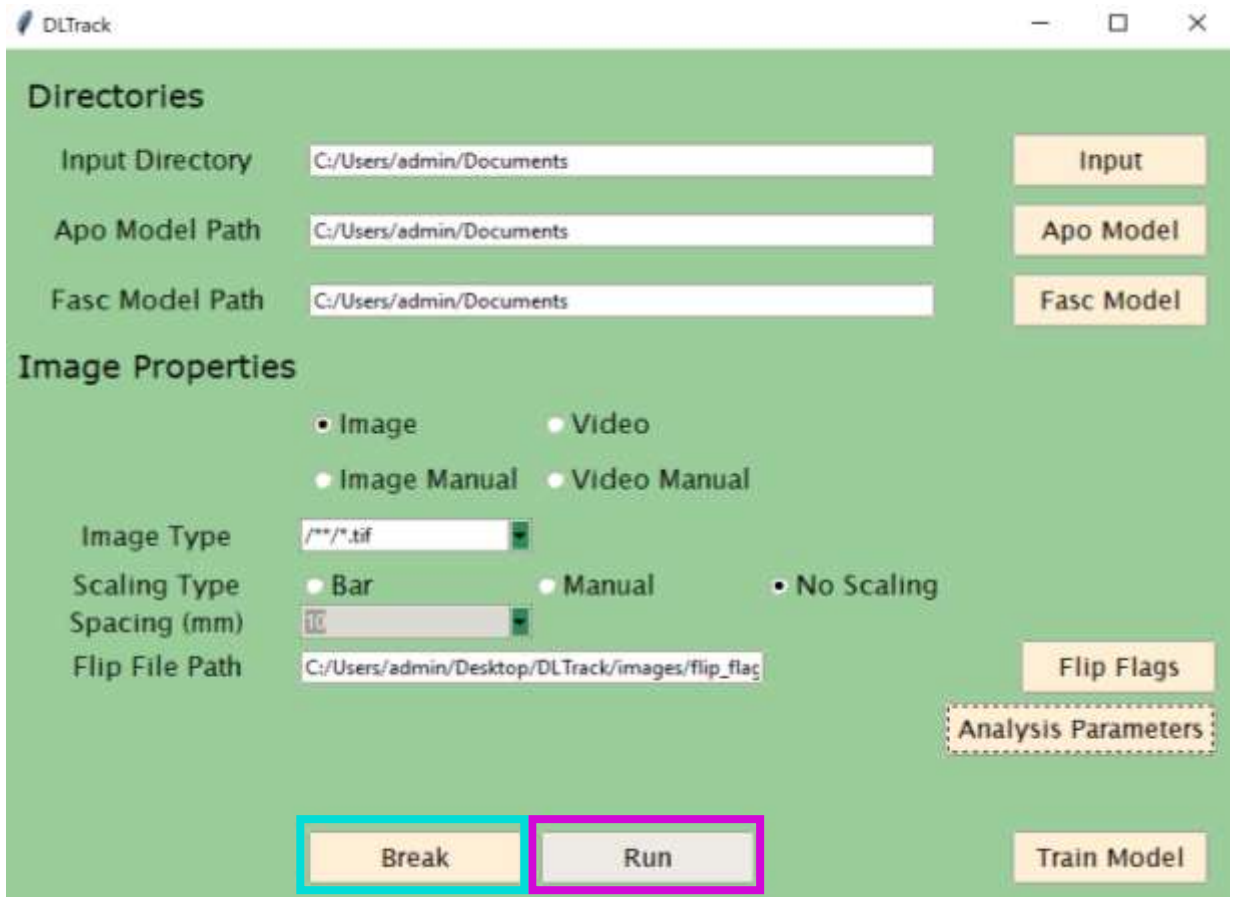
In the **Analysis Parameter window**, all parameters used by the **aponeurosis and fascicles neural networks** during inference are specified. We will give a short explanation to each of those parameters. The **Apo Threshold** parameters determines the threshold of the minimal acceptable probability by which a pixel is predicted as aponeurosis. The lower, the more pixels will be classified as aponeurosis. The **Fasc Threshold** is the same thing just for fascicle segments. The lower the **Fasc Cont Threshold**, the shorted the minimal acceptable length of detected fascicle segments to be included in the results. The **Minimal Width** determined the minimal acceptable distance between superficial and deep aponeurosis. **Minimal and Maximal Pennation** describe the respective minimal and maximal pennation angle that is physiologically possible in the analysed image/muscle.

For this example, you can leave all parameters the way they are. You can set the parameters by clicking the **Set parameters** button, the **Analysis Parameter window** will then close automatically. Please make sure to adapt these parameters according to your images in analyses outside of this example. For future analyses, it's best you test the ideal parameter configuration to get the best prediction results on your images in a small sample prior to the actual analysis.



## 6. Running / Breaking DL\_Track

Now its time to start the actual analysis of the example images in the “DL\_Track\_example/images” folder. You can do so by clicking the **Run** button in the main GUI window.



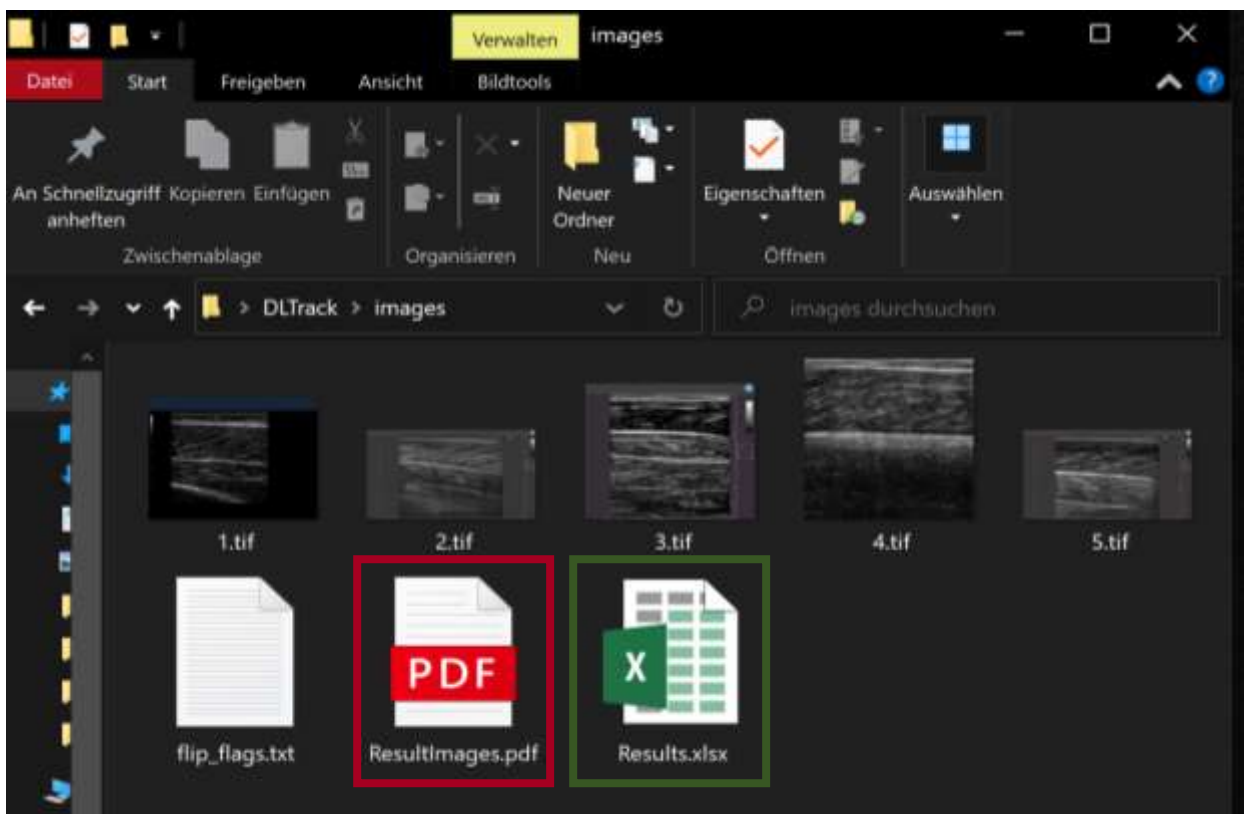
When you started the GUI using the command prompt in any way, you will see that the analysis is started by the statements printed in the prompt. When you started the GUI using the executable, you will just have to believe us that the analysis is started.

Moreover, you can see that there is a **Break** button placed in the GUI as well. Clicking the **Break** button allows you to stop the analysis at any point. The currently evaluated image will be processed and then the analysis is terminated.

Take a look at the next page to see what happens when the analysis is finished.



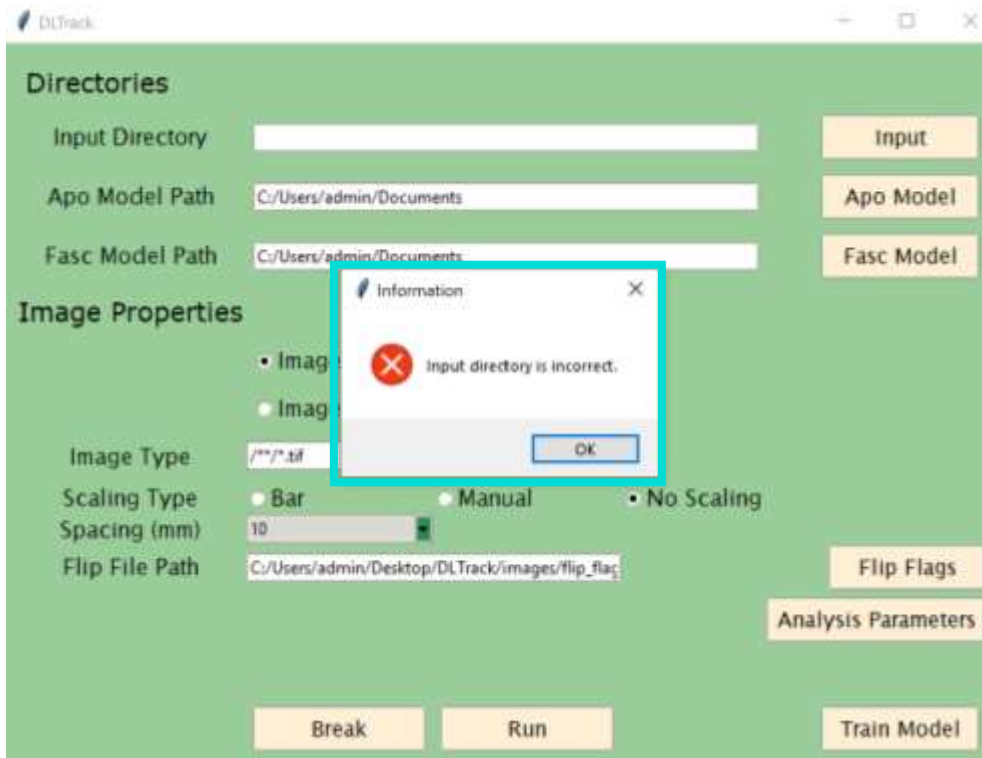
Subsequently to clicking the Run button in the main GUI, navigate again to the “DL\_Track\_example/images” folder in your explorer. You will see that two files will be / have been created, **ResultImages.pdf** and **Results.xlsx**. The **ResultImages.pdf** file contains each original input image and concomitant prediction results with fascicles and aponeurosis displayed. This file allows you to visually inspect the model outputs. In your future analysis outside of this tutorial, you should always visually inspect the **ResultImages.pdf** file. The **Results.xlsx** file contains the actual architectural parameter estimates for each input image. There, the median value of all detected muscle fascicle length and pennation angles as well as the calculated muscle thickness will be displayed. Each input image is displayed in a separate row. Note that the **ResultImages.pdf** file can be opened only after the **Results.xlsx** was created.



When both files can be opened and you can see the analysis results, original image and the prediction result, we must congratulate you! You have now officially and successfully completed the DL\_Track tutorial for automated image analysis! There is one more thing though, error handling. Take a look at the next section to get more information.

## 7. Error handling

Whenever an error occurs during the analysis process, the DL\_Track GUI will open a **messagebox**. This looks always similar to this:



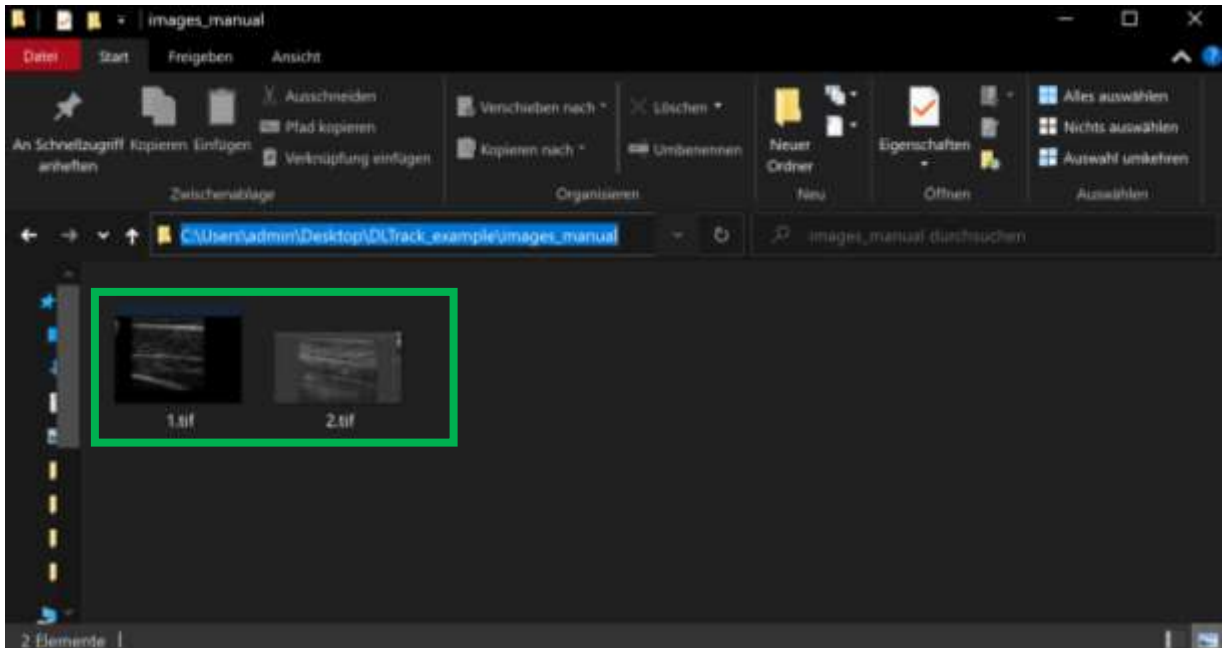
We tried to formulate these **messageboxes** as concise as possible. Just follow their instructions to fix the error and run the analysis anew. In case an error occurs that is not caught by an error **messagebox**, don't hesitate to report this in the Q&A section in the [DL\\_Track discussion forum](#). Please take a look at the "DL\_Track\_bugreport.md" file in this folder how do best do this. Otherwise, you can contact us by email at [paul.ritsche@unibas.ch](mailto:paul.ritsche@unibas.ch), but we would prefer the other way.

# *Manual Image Analysis*

The DL\_Track python software package offers several different analysis types for analysis of human lower limb longitudinal ultrasonography images. The next analysis type this tutorial covers is the manual image analysis. The images are evaluated manually by drawing the muscle thickness, fascicle length and pennation angles directly on the Image. Scaling the images will ensure estimated muscle architectural parameters are converted to centimetre units. For this type of analysis, single images (not videos) are a prerequisite. These images should be contained in a single folder, like in the “DL\_Track\_examples/images\_manual” folder. If you haven’t downloaded this folder, please do so now (link: [https://github.com/DFG-Lab/DL\\_Track/blob/master/examples/images\\_manual](#)). Unzip the folder and put it somewhere accessible, for example on your desktop. We will make use of the included example files extensively during this tutorial. In the next few pages, we will look at every required step to successfully perform automated image analysis with DL\_Track.

# 1. Creating image directory

In order for DL\_Track to recognize your images, they should best be in a single folder (though one subfolder structure is acceptable as well). Take a look how you might structure this:

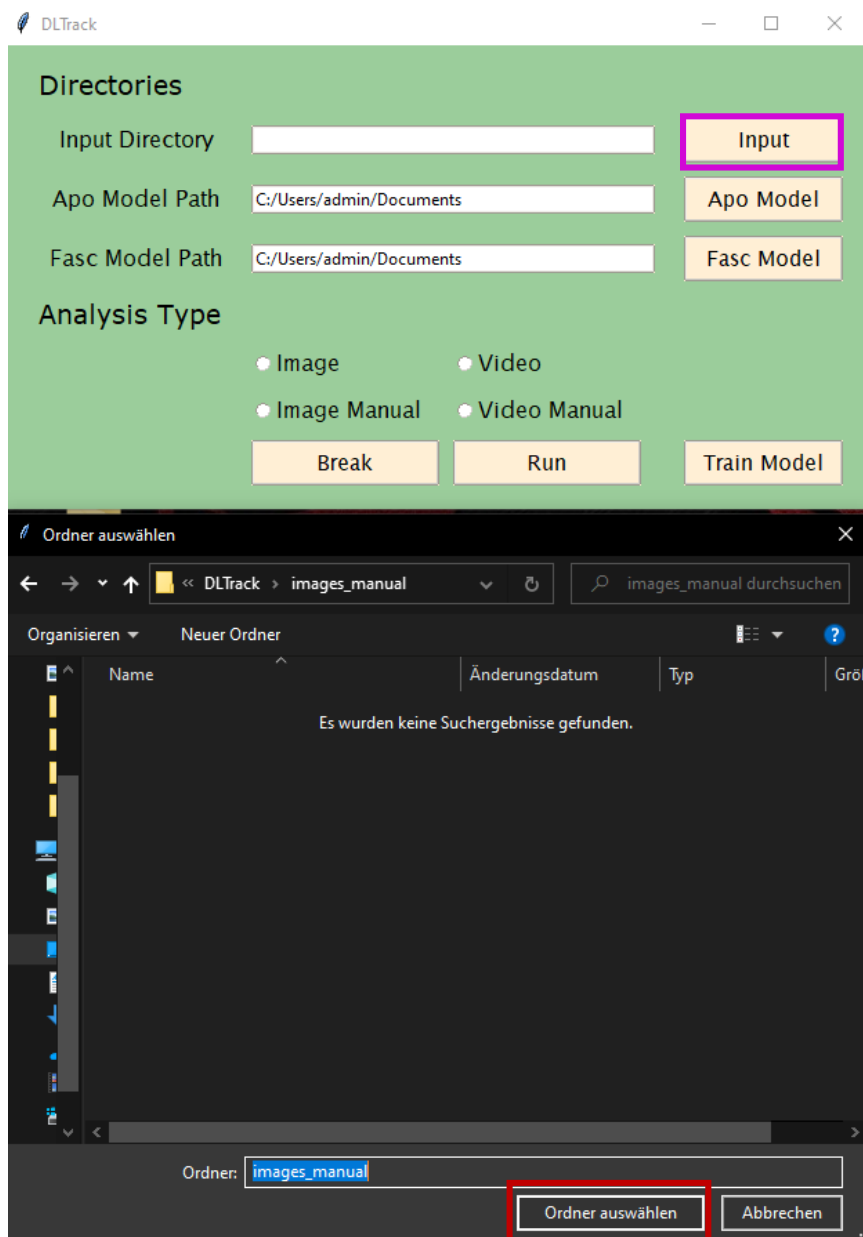


You can see in the picture above that the folder contains **2 images** and is **located on the desktop**. This structure is already included in the “DL\_Track\_example” folder. In contrast to automated image analysis, you do not need a flip\_flag.txt file nor do you need neural networks that do predictions. Here, you are the neural network. So, the next step is to specify the input directory in the GUI.

## 2. Specifying Input Directories in the GUI

You can start with the actual analysis! The first step of every analysis type in DL\_Track is to specify the input directories in the graphical user interface (GUI). We assume that you have already opened the GUI. (If not, take a look at the first chapter of this document “Starting the GUI”).

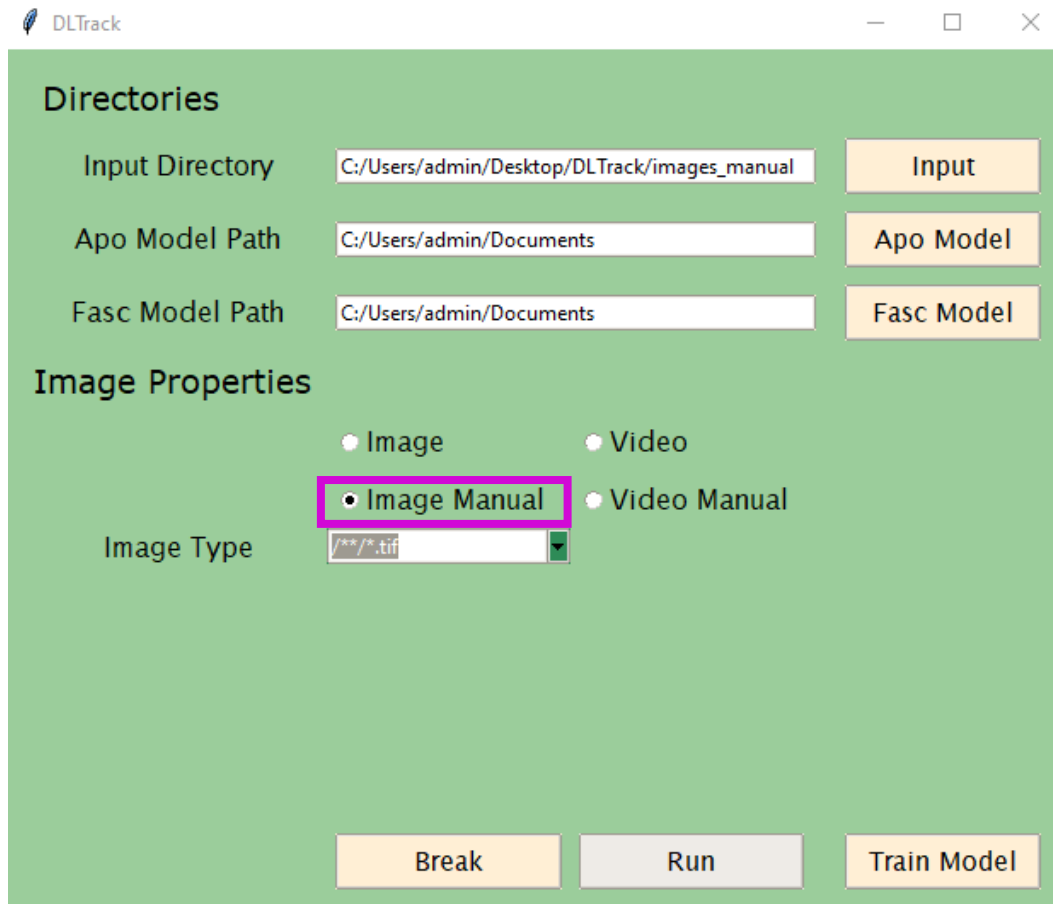
You will begin with specifying the path to the folder containing the **images** to be analysed. Remember this is the folder “DL\_Track\_example/images\_manual”. By clicking on the **Input** button in the GUI a selection window opens where you need to select the images folder. Click **select folder** to specify the path in the GUI.



Once that is done, the path will be displayed in the entry field and you can start to specify the relevant parameters for the analysis.

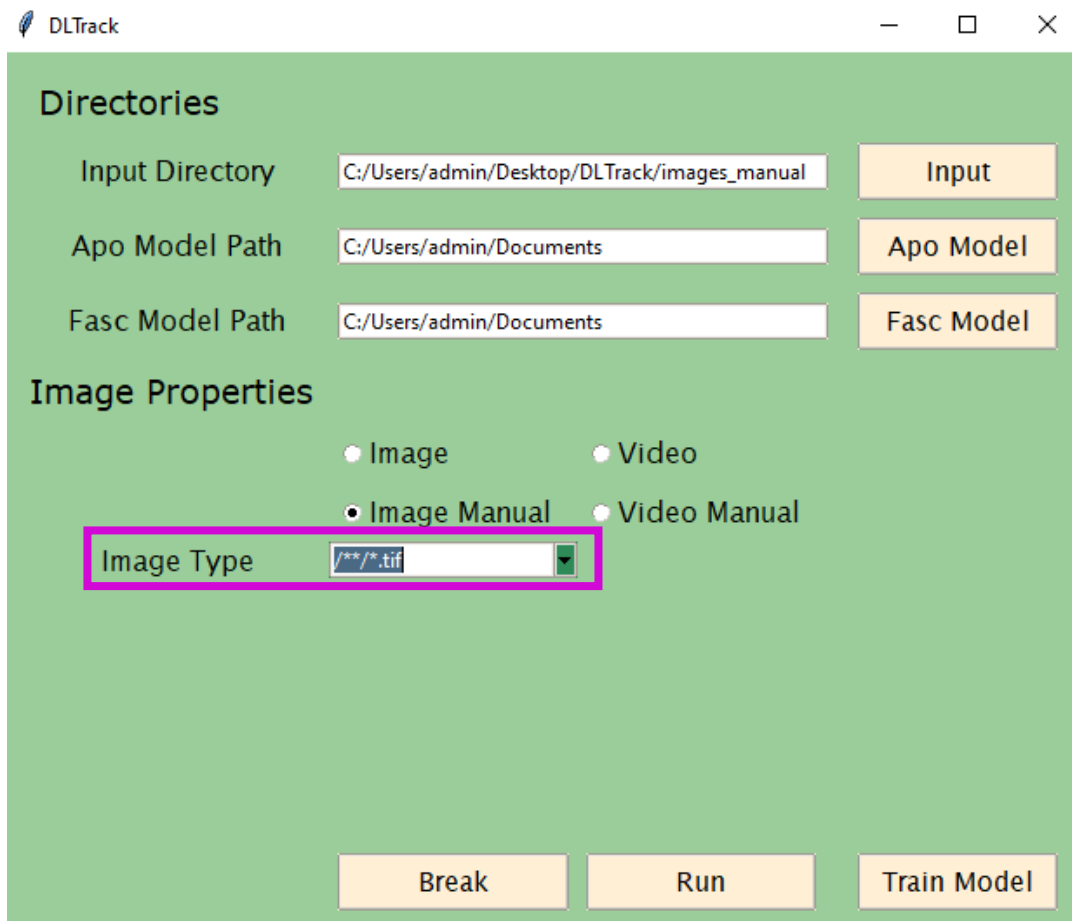
### 3. Specifying Relevant Parameters

As a first step, you will select the right analysis type in the GUI. Since this section is about manual image analysis, please select the **Image Manual** radiobutton. You can see that the GUI unfolds and another parameter appear. You will set this one in the next step on the next page.



The screenshot shows the DLTrack GUI with a green background. At the top, the title bar says 'DLTrack' with standard window controls. The main area is divided into two sections: 'Directories' and 'Image Properties'. In the 'Directories' section, there are three rows: 'Input Directory' with a text box containing 'C:/Users/admin/Desktop/DLTrack/images\_manual' and an 'Input' button; 'Apo Model Path' with a text box containing 'C:/Users/admin/Documents' and an 'Apo Model' button; and 'Fasc Model Path' with a text box containing 'C:/Users/admin/Documents' and a 'Fasc Model' button. The 'Image Properties' section has four radio buttons: 'Image', 'Video', 'Image Manual' (which is selected and highlighted with a pink box), and 'Video Manual'. Below these is a label 'Image Type' followed by a dropdown menu showing '/\*.\*.tif'. At the bottom, there are three buttons: 'Break', 'Run' (disabled), and 'Train Model'.

Next, you need to specify the Image Type. The ending of the **Image Type** must match the ending of your **images**, otherwise no files are found by DL\_Track. You can either select a pre-specified ending from the dropdown list or type your own ending. Please keep the formatting similar to those **Image Types** provided in the dropdown list. All the **images** in the “DL\_Track\_example/images\_manual” folder are of the **Image Type** “.tif”. Thus, you should select the “/\*\*/\*.tif” **Image Type**.



The image shows a screenshot of the DLTrack application window. The window has a title bar with the DLTrack logo and standard window controls. The main content area is divided into two sections: "Directories" and "Image Properties".

**Directories Section:**

- Input Directory:** A text box containing "C:/Users/admin/Desktop/DLTrack/images\_manual" with an "Input" button to its right.
- Apo Model Path:** A text box containing "C:/Users/admin/Documents" with an "Apo Model" button to its right.
- Fasc Model Path:** A text box containing "C:/Users/admin/Documents" with a "Fasc Model" button to its right.

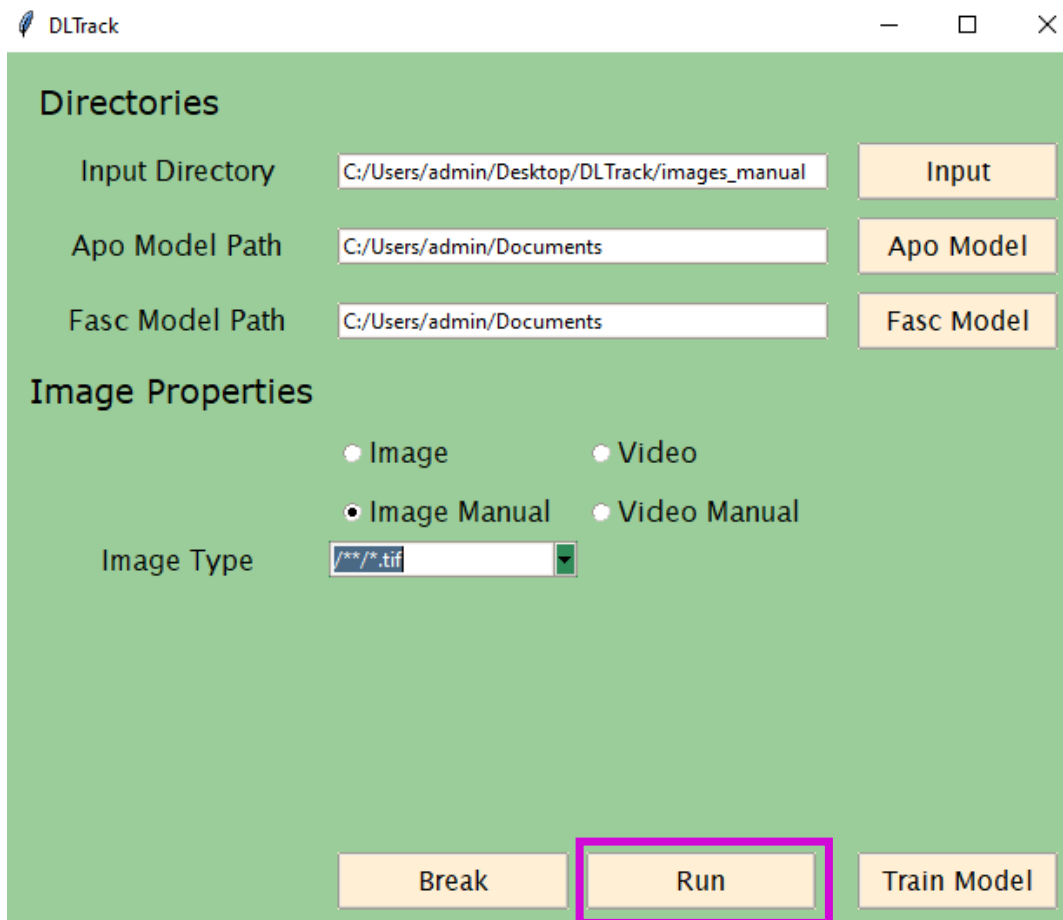
**Image Properties Section:**

- Four radio buttons are present: "Image", "Video", "Image Manual", and "Video Manual". The "Image Manual" option is selected.
- Image Type:** A dropdown menu is highlighted with a red rectangle, showing the selected value "/\*\*/\*.tif".

**Buttons:** At the bottom of the window, there are three buttons: "Break", "Run", and "Train Model".

Allright, once you have specified the Image Type, you can start with the analysis of the images contained I the “DL\_Track\_example/images\_manual” folder.

Allright, once you have specified the Image Type, you can start with the analysis of the images contained I the “DL\_Track\_example/images\_manual” folder. You can start the analysis by clicking the **Run** button in the main GUI.

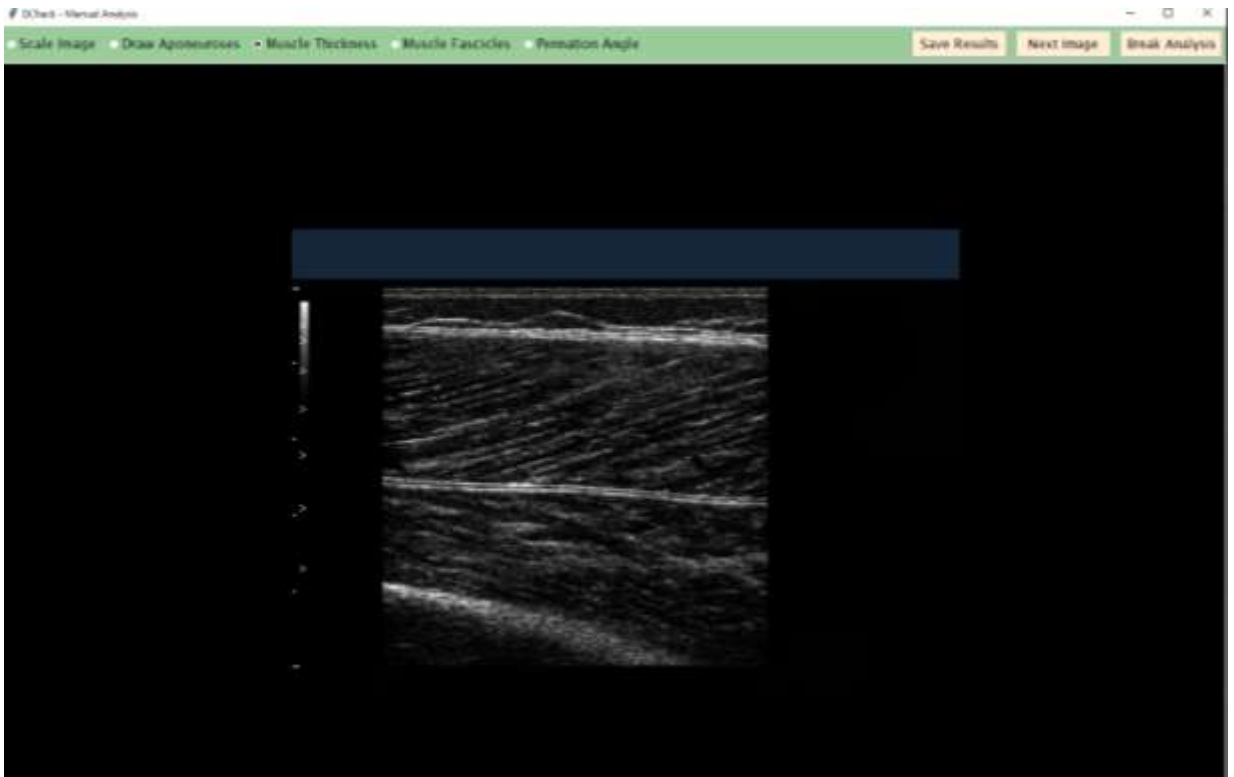


Take a look at the next page to see how to continue in the “Manual Analysis window” that pops up.



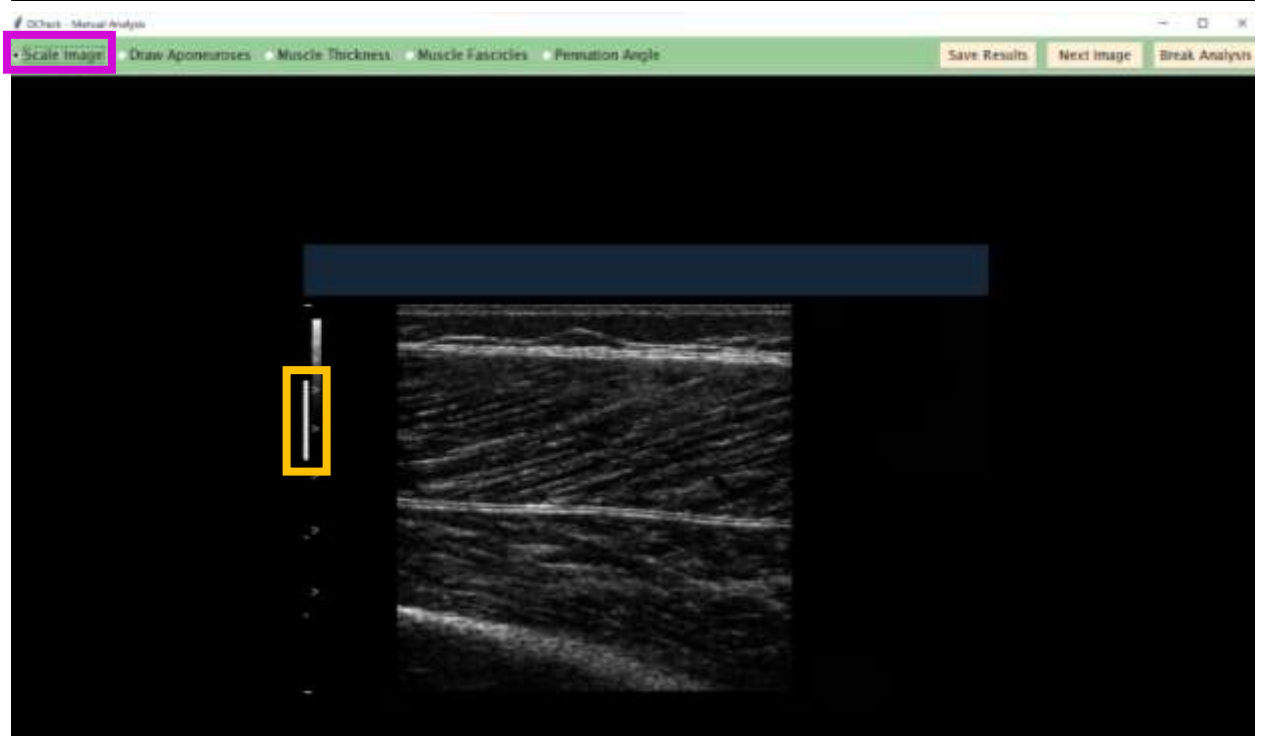
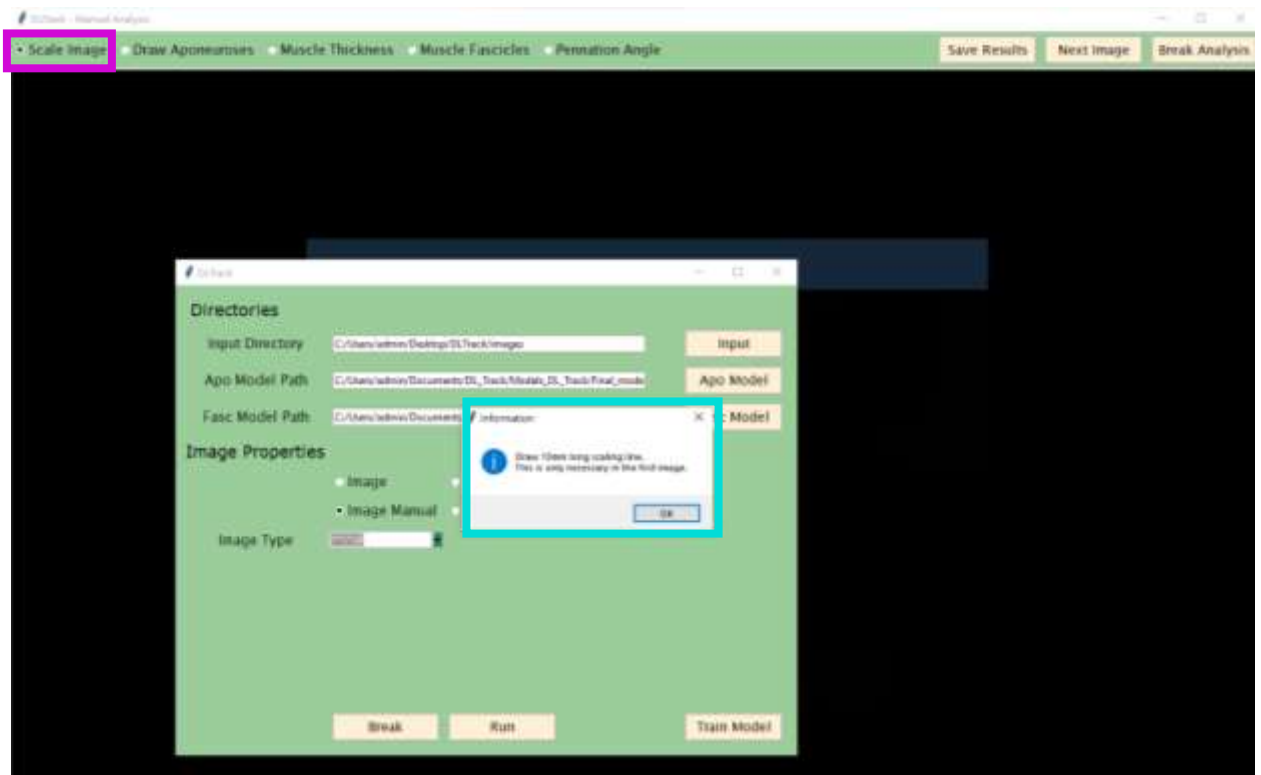
## 4. Manual analysis of images

Subsequent to clicking the Run button in the main GUI, the “Manual Analysis window” opens. Here you can analysis the image by marking the respective architectural parameter you want to analyse directly on the image. We will guide you through that on the next pages. But first of all, here is how the “Manual Analysis window” looks like:

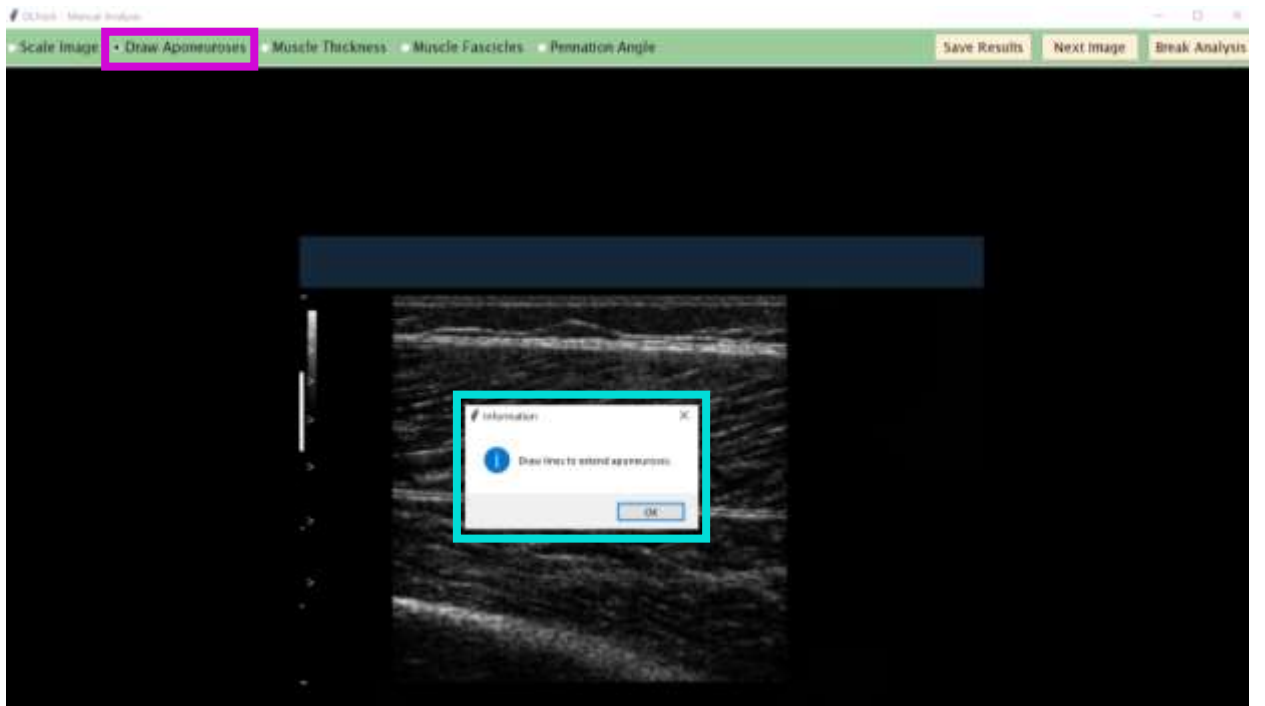


It is of utmost importance to keep in mind that the lines you are about to draw on the image are only there to help the human mind remember what it did. The actual lines are not used during the computation of the architectural parameters. The start- and endpoints of each line are relevant. The start point is defined as the point where you clicked the left mouse button to start drawing the line. The endpoint is defined as the point where you released the left mouse button to stop drawing the line. The line follows the cursor as long as the left mouse button is pressed. The calculations of the scaling line length, muscle thickness, fascicle length and pennation angle are dependent on the number of specified lines/segments. Do NOT click somewhere random on the image during the analysis of a parameter and exactly follow the instructions. If additional clicks happened, start the analysis anew by selecting the radiobutton representing the parameter again. If you do not follow the instructions presented in this tutorial, we cannot guarantee the correctness of the analysis results.

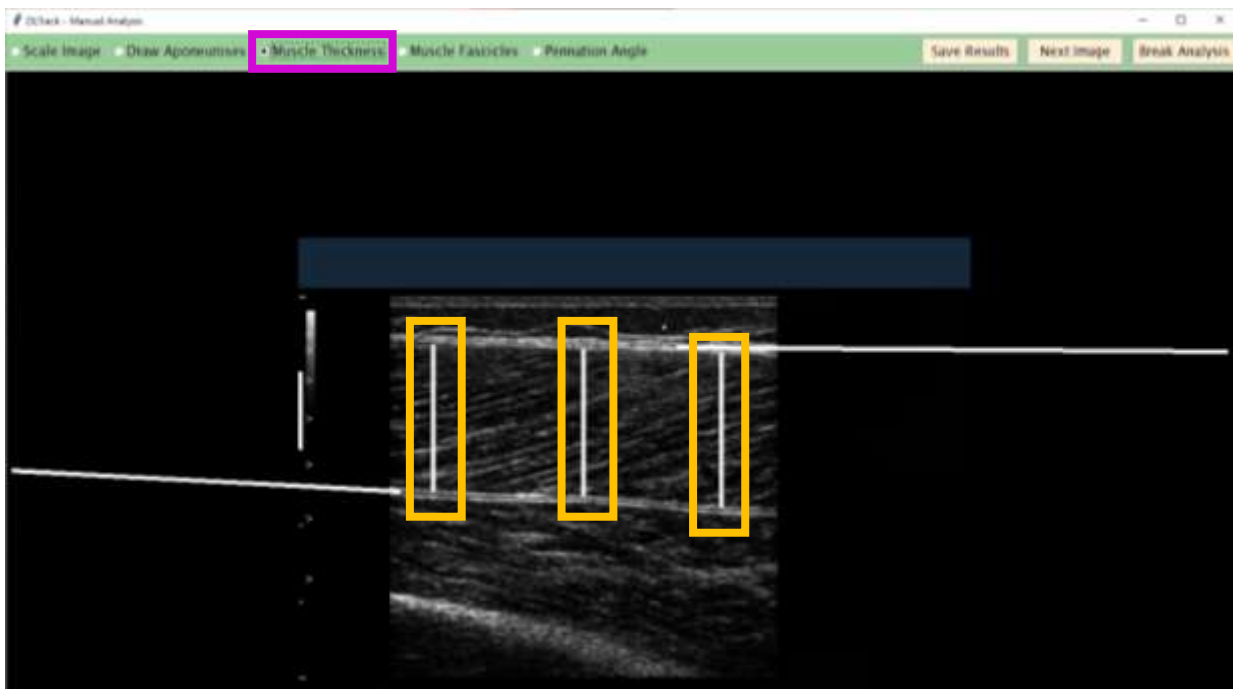
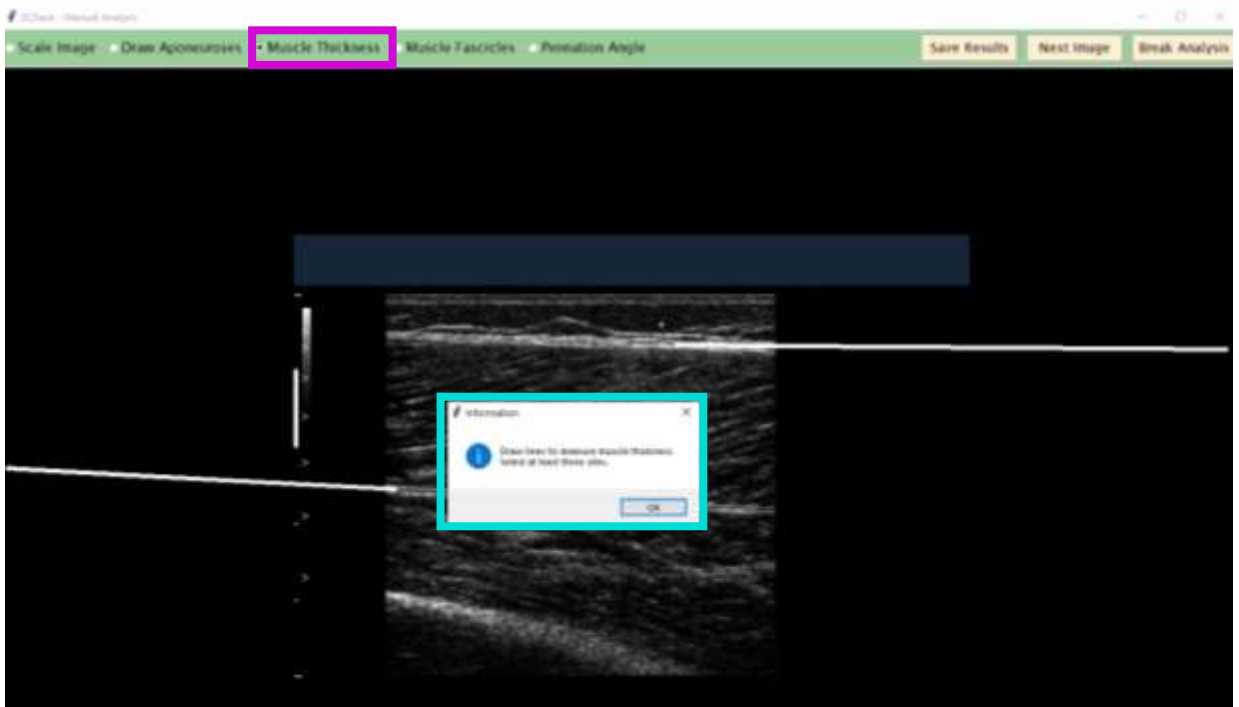
First of all, you will scale the images manually so that the calculated architectural parameters are returned in centimetre rather than pixel units. For the scaling, please draw a **one centimetre long straight line in the image**. The distance of one centimetre is usually recognizable in the scaling bars in the image. You can initiate the scaling process by selecting the **Scale Image** radiobutton in the “Manual Analysis window”. A **messagebox** will appear advising you what to do. The **drawn line** should look like this.



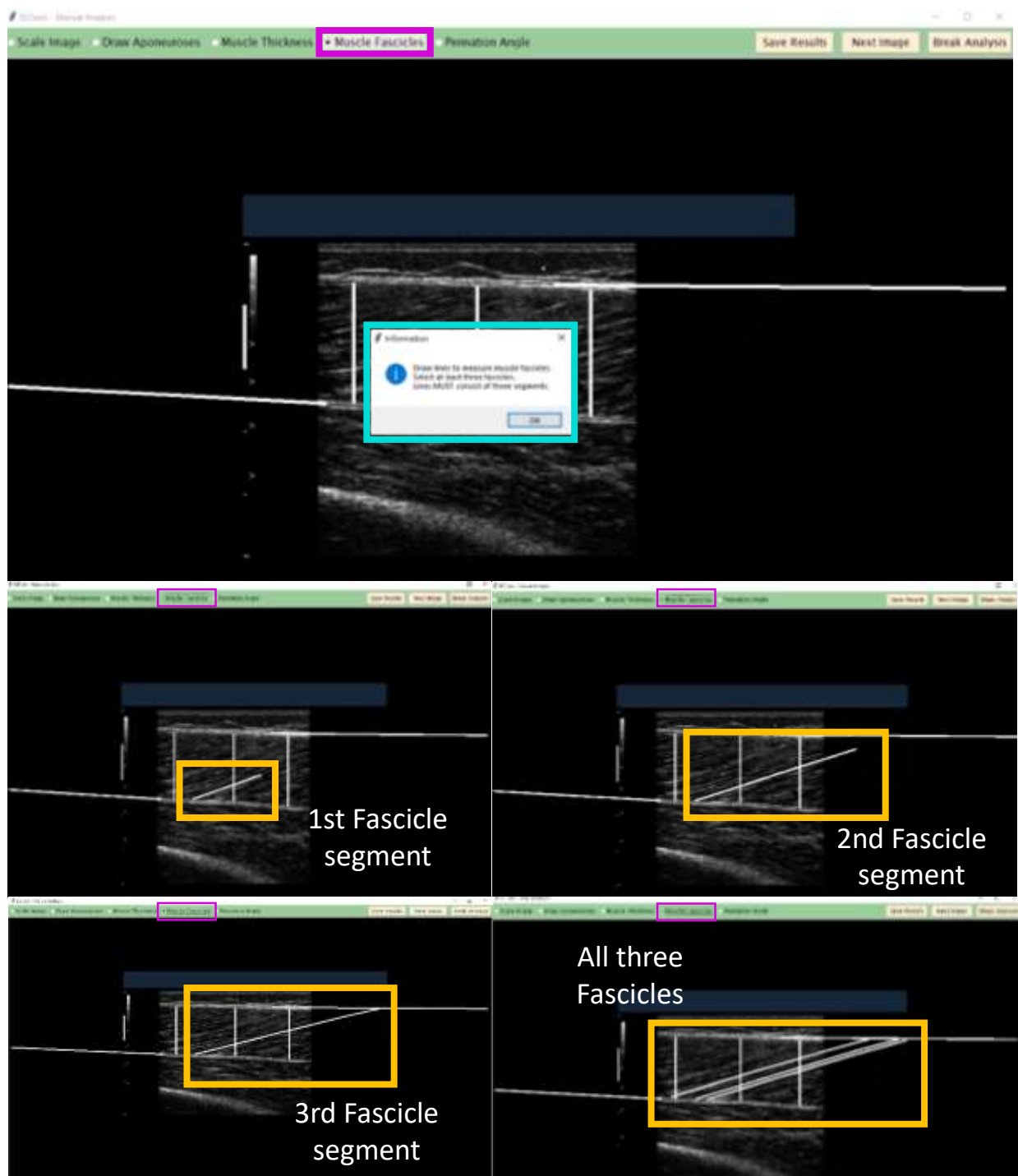
As a next step you have the option to extend the muscle aponeuroses to ease the extrapolation of fascicles extending outside of the image. This however not required, merely an option. You can do by selecting the **Draw Aponeurosis** button in the “Manual Analysis window” and draw the **aponeurosis lines** on the image as shown below. A **messagebox** will appear advising you what to do.



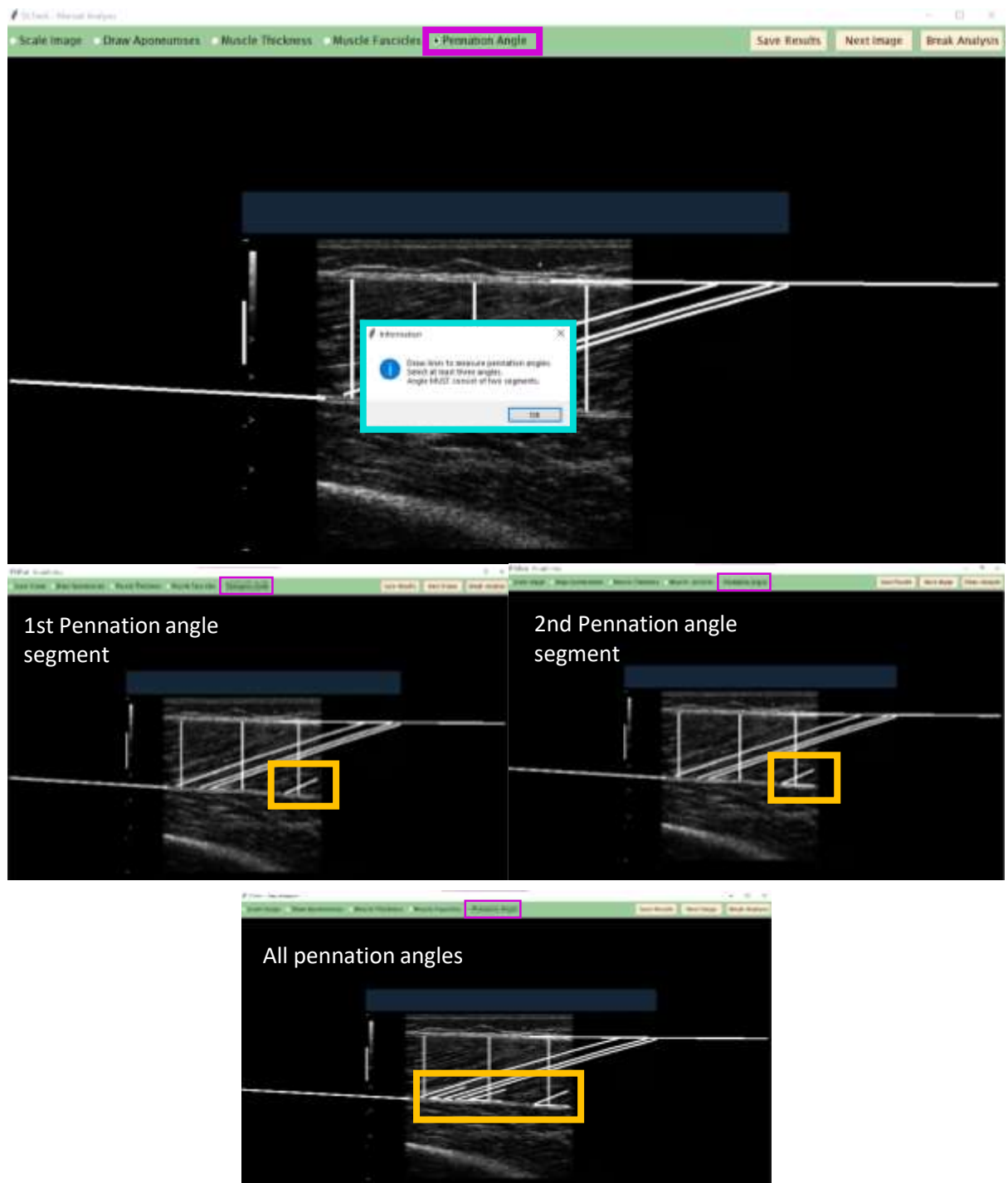
Once the image is scaled and aponeuroses structures extended, you can start analysing the muscle architectural parameters. You will start with the muscle thickness. Therefore, select the **Muscle Thickness** radiobutton in the “Manual Analysis window”. A **messagebox** will appear advising you what to do. We advise to you to now draw **three straight lines** reaching from the superficial to the deep aponeurosis in the middle, right and left portion of the muscle image.



Next you can mark single fascicles on the image by selecting the **Muscle Fascicles** radiobutton in the “Manual Analysis window”. A **messagebox** will appear advising you what to do. We advise to draw at least three **fascicles** per image in different regions of the image. Please consider that these fascicles must be clearly visible and not guessed. It is possible to extrapolate the **fascicles** outside of the image region. Each **fascicle** MUST consist of three segments. Do not draw more or less segments per **fascicle** and pay attention to avoid any extra unwanted mouse clicks. One segment MUST start where the previous segment ended. Take a look at the image sequence below to see how it is done:

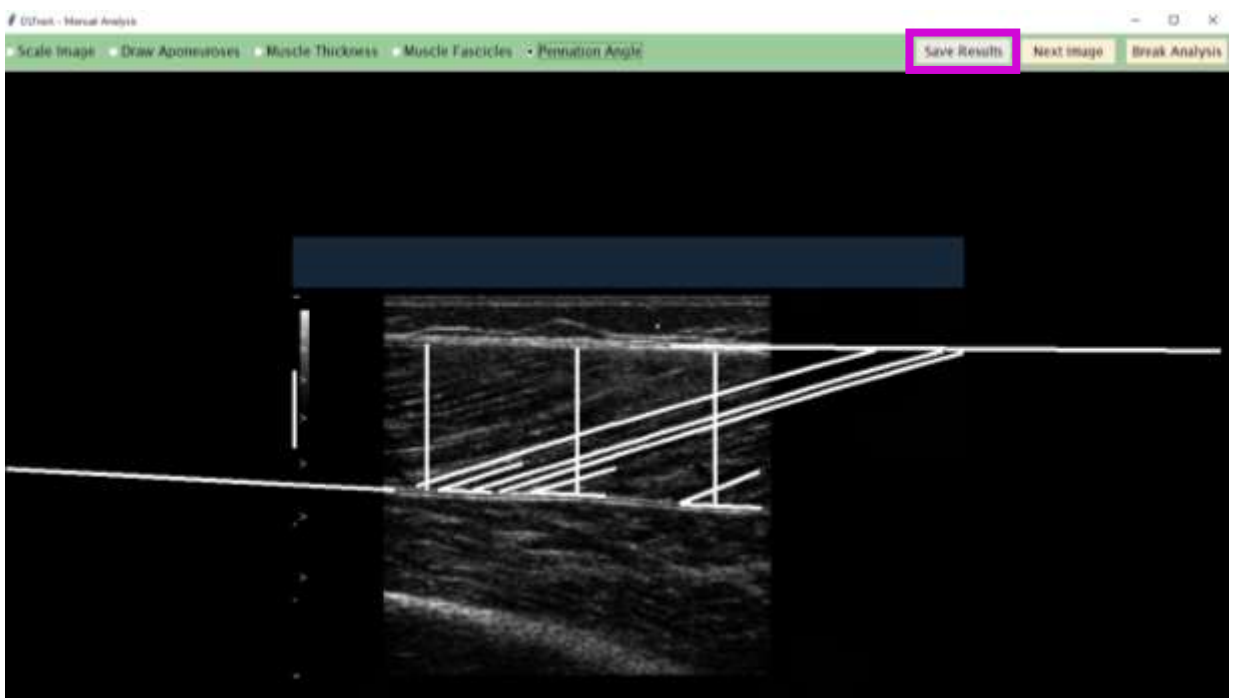


The last architectural parameter that you can manually analyse using the DL\_Track package is the muscle pennation angle. Please select the radiobutton **Pennation Angle**. A **messagebox** will appear advising you what to do. We advise you to draw at least three **pennation angles** per image at different regions of the image. As with the fascicles, these should be clearly visible. Each drawn **pennation angle** MUST consist of two segments. The first segment should follow the orientation of the fascicle, the second segment should follow the orientation of the deep aponeurosis. The segments should both originate at the insertion of the fascicle in the deep aponeurosis. Please pay attention to avoid unwanted clicks on the image. Here is how its done:



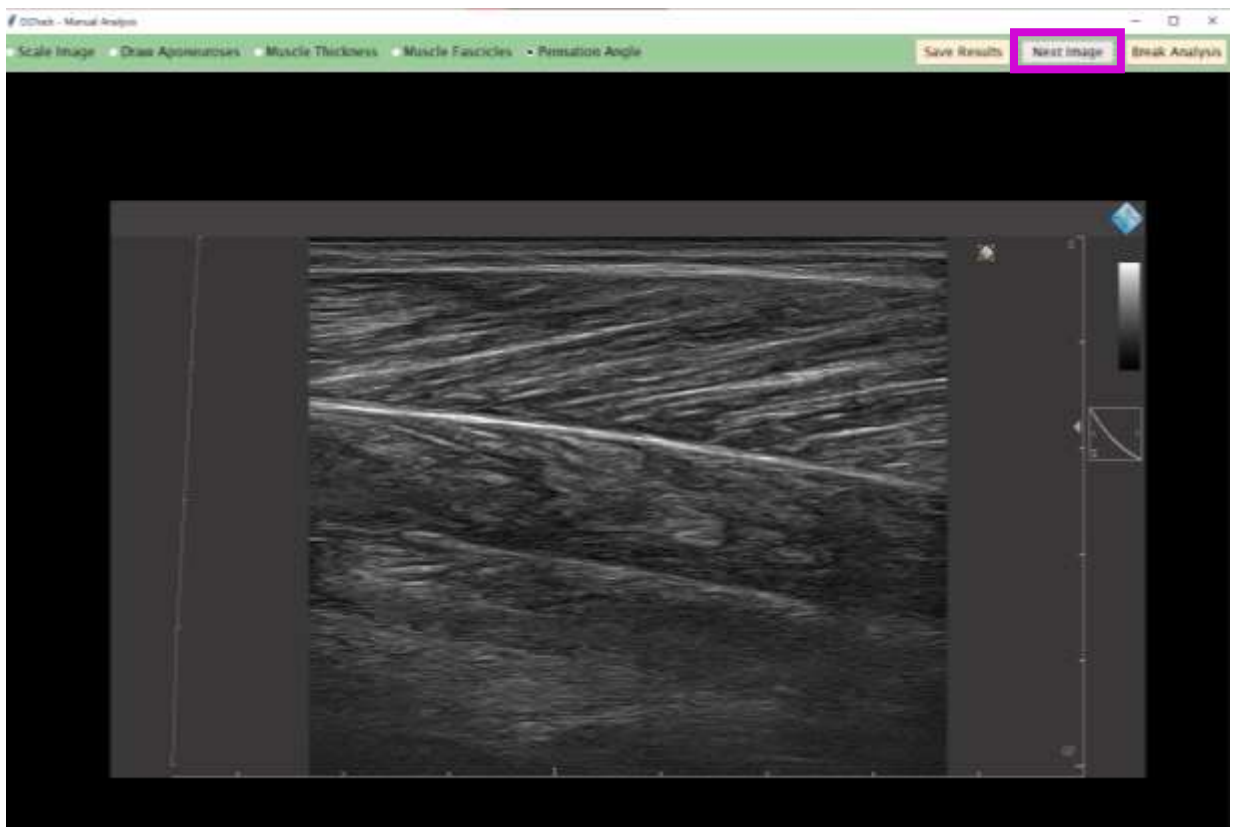
## 5. Saving / Breaking / Next Image

Great, you have now successfully analysed all muscle architectural parameters using the DL\_Track manual image analysis tool. However, there are three buttons in the “Manual Analysis window” left to explain. The first button is the **Save Results** button. The **Save Results** button is a very important button! Before you have pressed this button, none of your analysis results are saved. Therefore, please press the **Save Results** button once you have analyzed all parameters that you wanted to analyze and **before** continuing with the next image. An excel file with the name Manual\_Results.xlsx is saved in the directory of the input images upon pressing the **Save Results** button. Therein, all analysis results are stored. In your case the file is saved in the “DL\_Track\_example/images\_manual” folder.



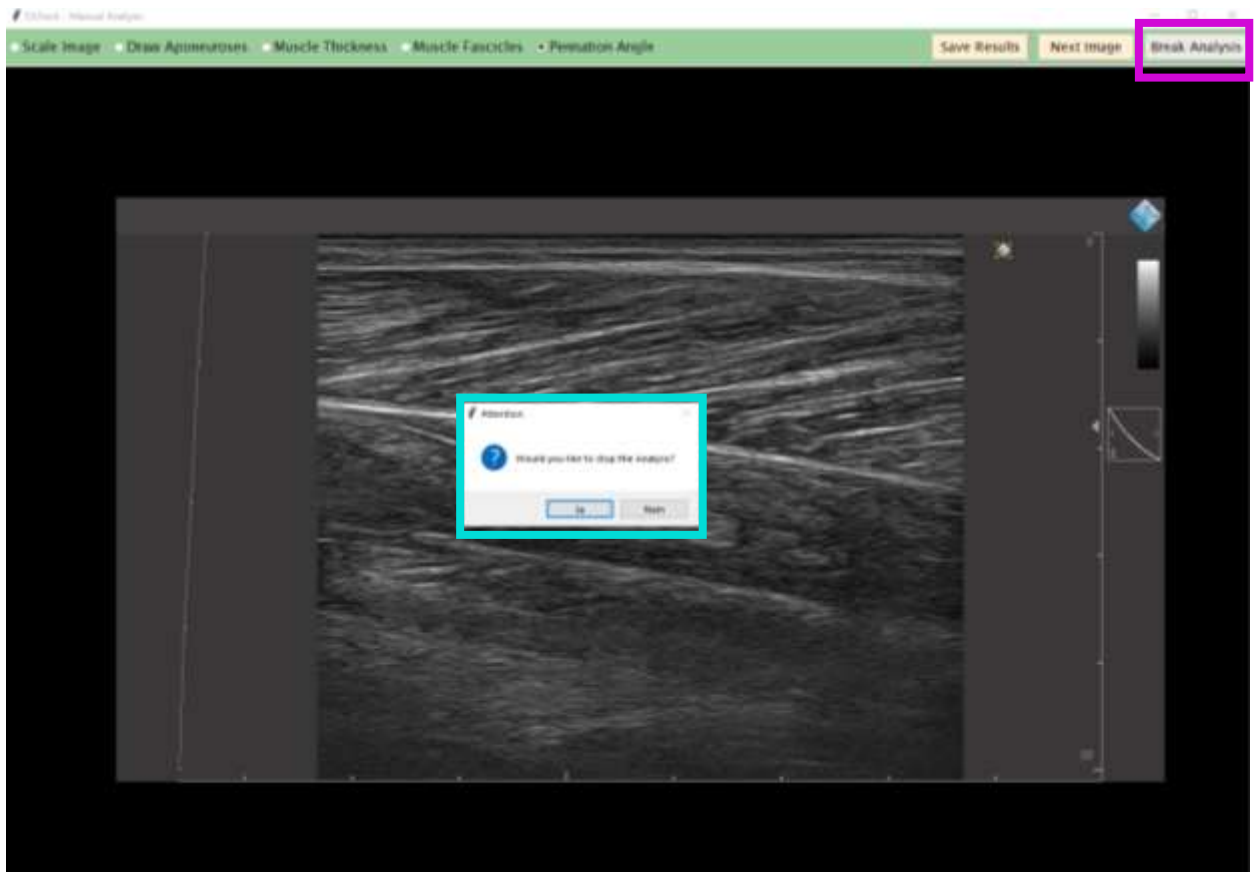


The second button we haven't explained yet is the **Next Image** button. By clicking this button, you can proceed to the next image in the input folder (in your case the "DL\_Track\_example/images\_manual" folder). Please remember to press the **Save Results** button prior to proceeding to the next images, otherwise your analysis results for this image will be lost. When the **Next Image** button is pressed, the displayed image is updated.

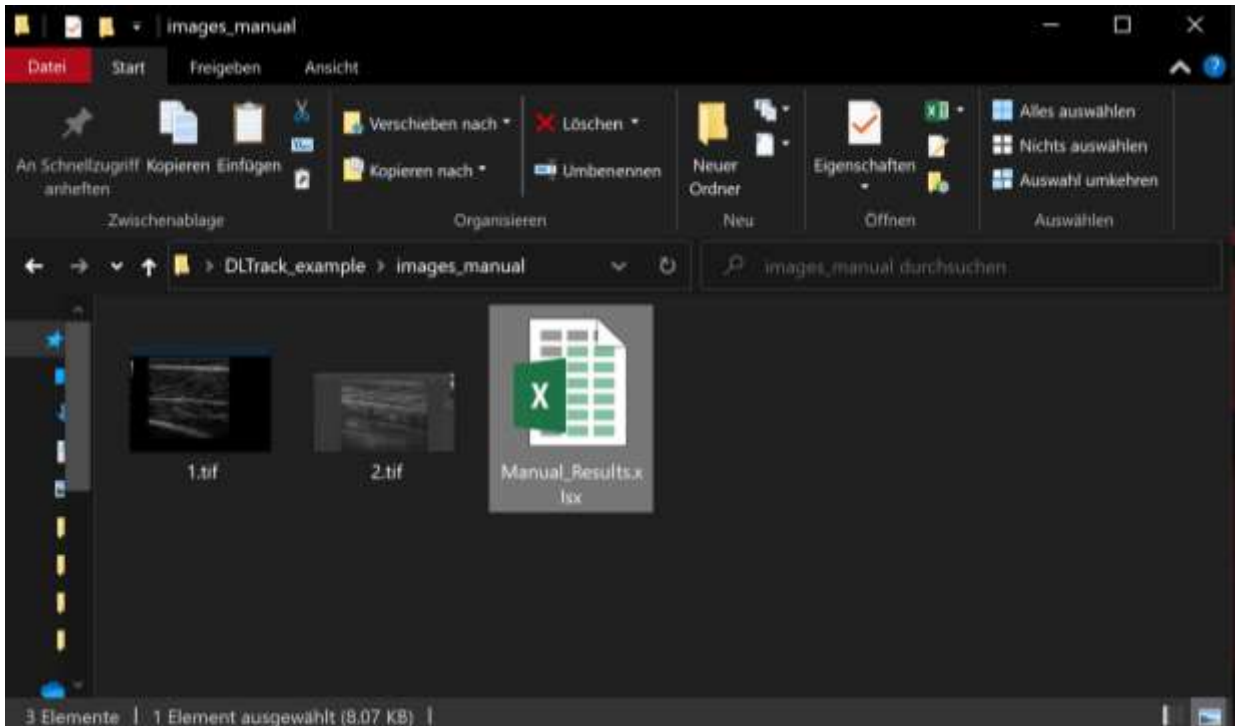




The last button we need to explain is the **Break Analysis** button. Pressing this button allows you to terminate the analysis and return to the main GUI window. A **messagebox** will appear asking you if you really want to stop the analysis. Once the **Break Analysis** button is pressed and you answered the messagebox with “YES”, the “Manual Analysis window” will be automatically closed. Remember to save your analysis results before doing so. We think you know which very important button can do just that!



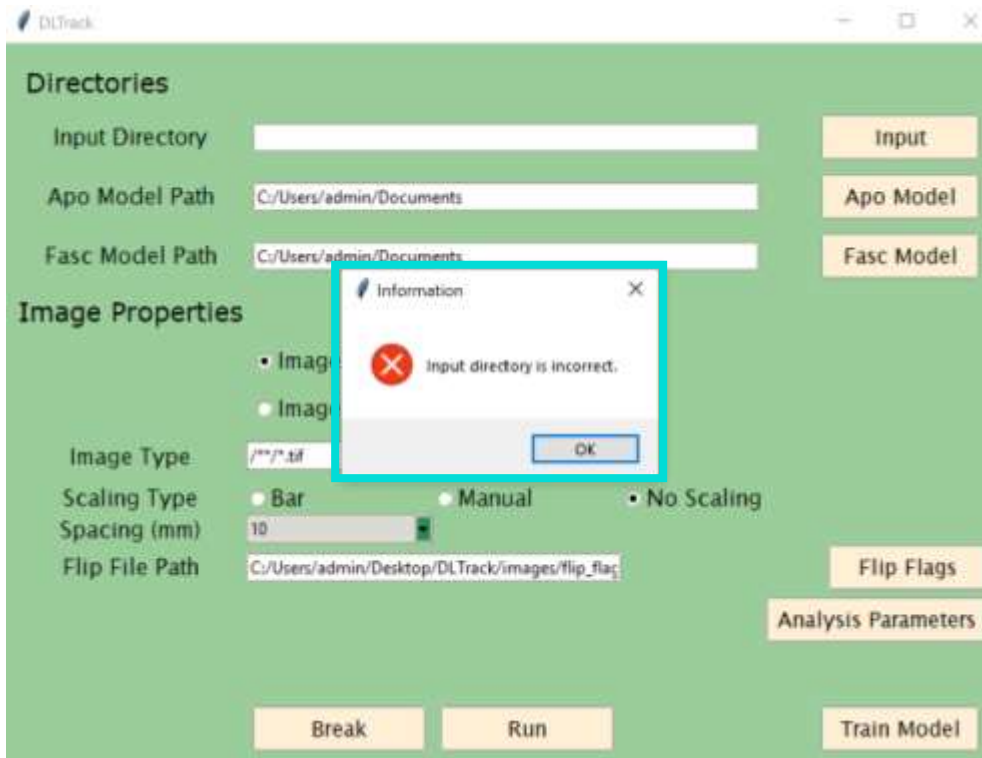
When you have saved your results clicking the very important button and followed our instructions during this tutorial, your input directory “DL\_Track\_example/images\_manual” should look like this. It should contain the images as well as the Manual\_Results.xlsx file.



Congrats are in order! You have now officially and successfully completed the DL\_Track tutorial for manual image analysis! There is one more thing though, error handling. Take a look at the next section to get more information.

## 6. Error handling

Whenever an error occurs during the manual image analysis process, the DL\_Track GUI will open a **messagebox**. This looks always similar to this:



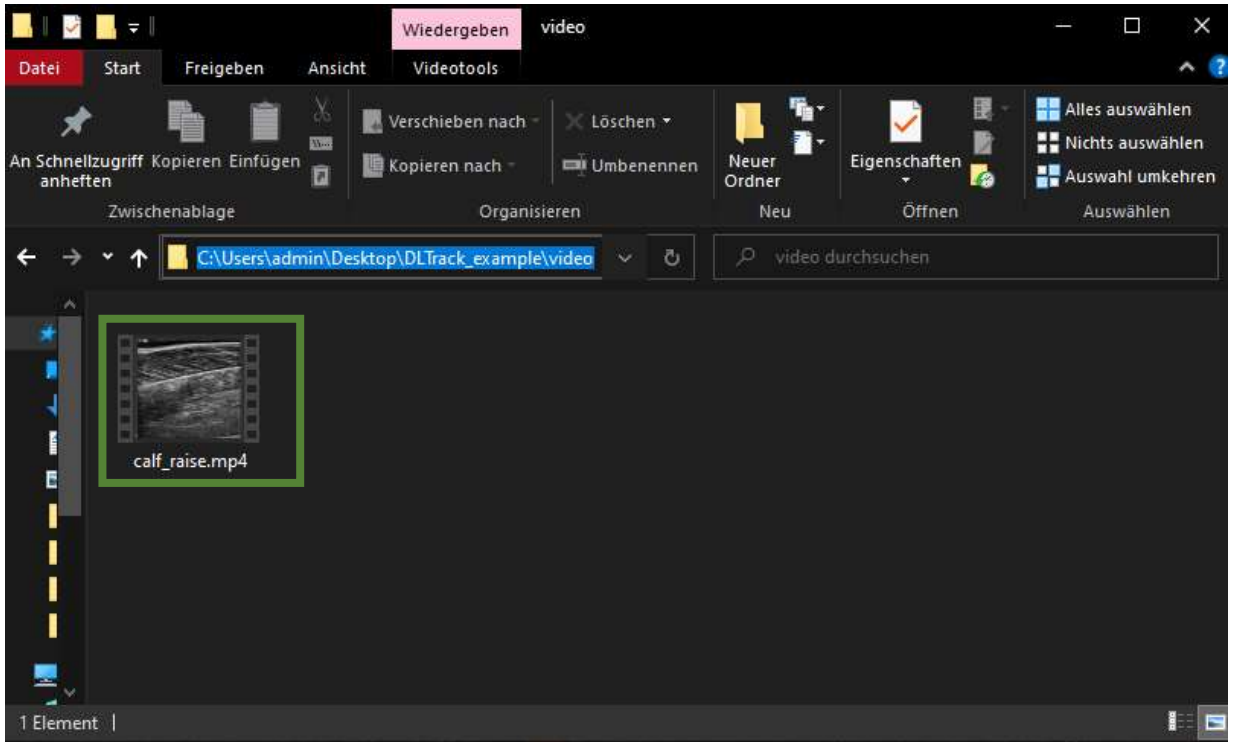
We tried to formulate these **messageboxes** as concise as possible. Just follow their instructions to fix the error and run the analysis anew. In case an error occurs that is not caught by an error **messagebox**, don't hesitate to report this in the Q&A section in the [DL\\_Track discussion forum](#). Please take a look at the "DL\_Track\_bugreport.md" file in this folder how do best do this. Otherwise, you can contact us by email at [paul.ritsche@unibas.ch](mailto:paul.ritsche@unibas.ch), but we would prefer the other way.

# ***Automated Video Analysis***

The DL\_Track python software package offers several different analysis types for analysis of human lower limb longitudinal ultrasonography images. This section of the tutorial covers the automated video analysis. The videos are evaluated without user input and may be scaled. Scaling the images will ensure estimated muscle architectural parameters are converted to centimetre units. For this type of analysis, videos are a prerequisite. These videos should be contained in a single folder, like in the “DL\_Track\_example/videos” folder. If you haven’t downloaded this folder, please do so now (link: [https://github.com/DFG-Lab/DL\\_Track/blob/master/example/videos](#)). Unzip the folder and put it somewhere accessible, for example on your desktop. We will make use of the included example files extensively during this tutorial. The automated video analysis is very similar to the automated image analysis. In fact, the inputted video is analysed frame by frame and each frame is therefore treated like an independent image. Moreover, only few analysis parameters are different between both analysis types. In the next few pages, we will look at every required step to successfully perform automated video analysis with DL\_Track.

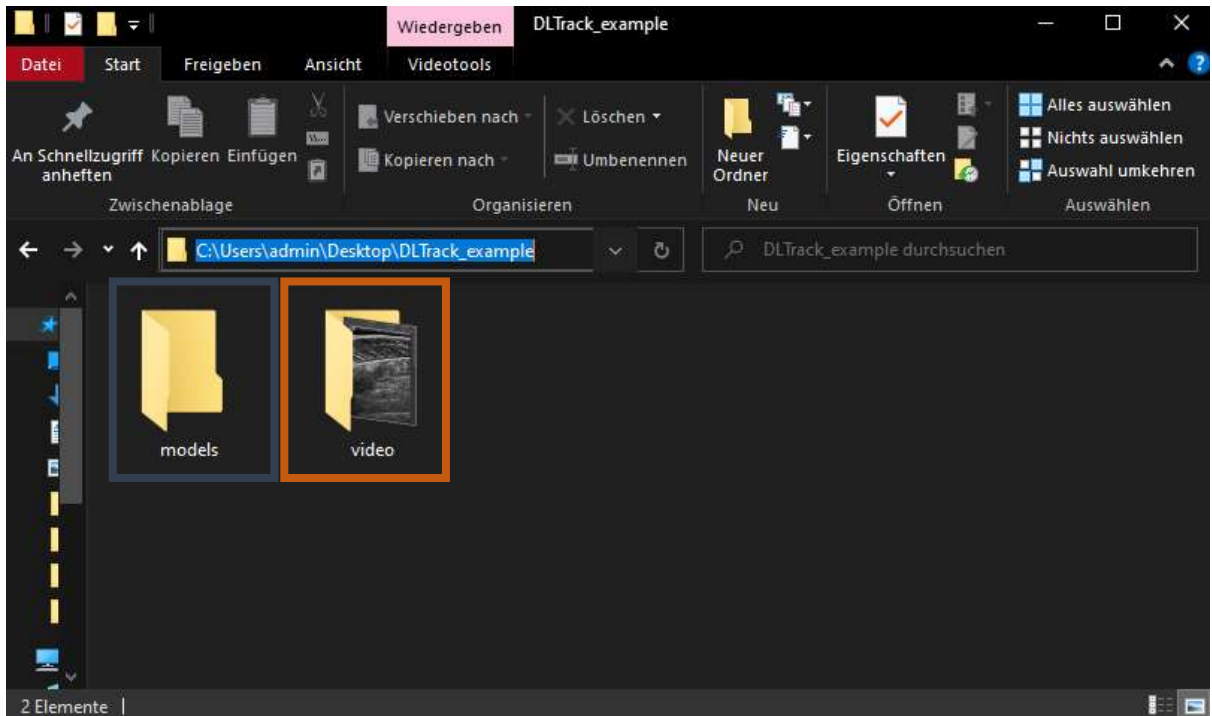
# 1. Creating video and network directories

In order for DL\_Track to recognize your videos, they should best be in a single folder (though one subfolder structure is acceptable as well). Take a look how you might structure this:



You can see in the picture above that the folder contains **one video** and is **located on the desktop**. This structure is already included in the “DL\_Track\_example” folder. We will continue with demonstrating how to create folders for the **aponeurosis and fascicle neural networks** on the next page.

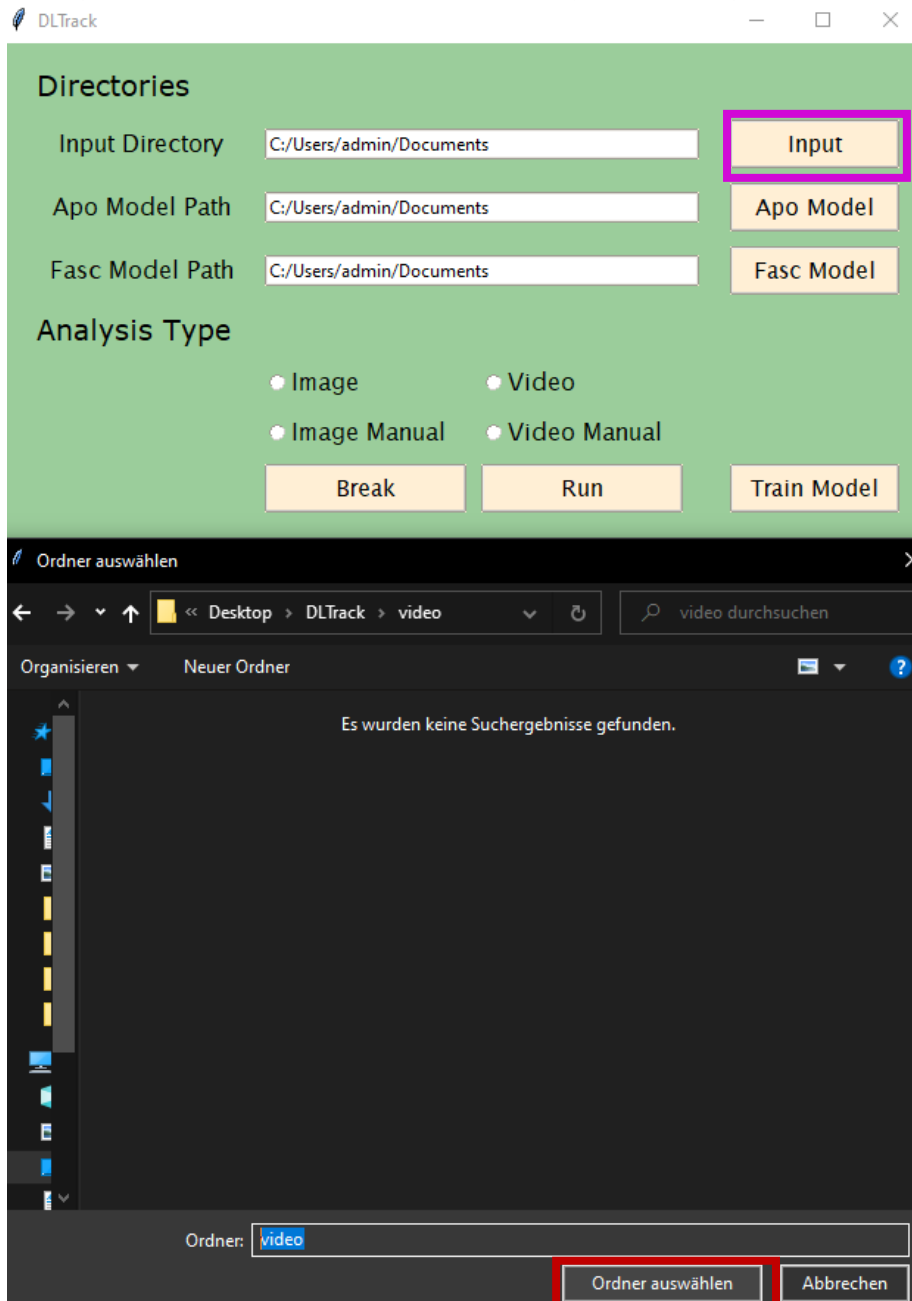
The **folder containing the video** (in this case the “DL\_Track\_example/video” folder) is already included in the “DL\_Track\_example” folder. We will now create a separate folder for the pre-trained **aponeurosis and fascicle neural networks**. In case you have not downloaded the **models**, please do so now (link: ). Place them in a subfolder of the “DL\_Track\_example” folder for the moment, like “DL\_Track\_example/**models**”. You will make use of these **neural networks** later as well, when you analyse your own images outside of this tutorial. Of course you can move them to a different folder then. Your folder structure inside the “DL\_Track\_example” folder should now look something like this with **the neural networks** placed in the **models** folder.



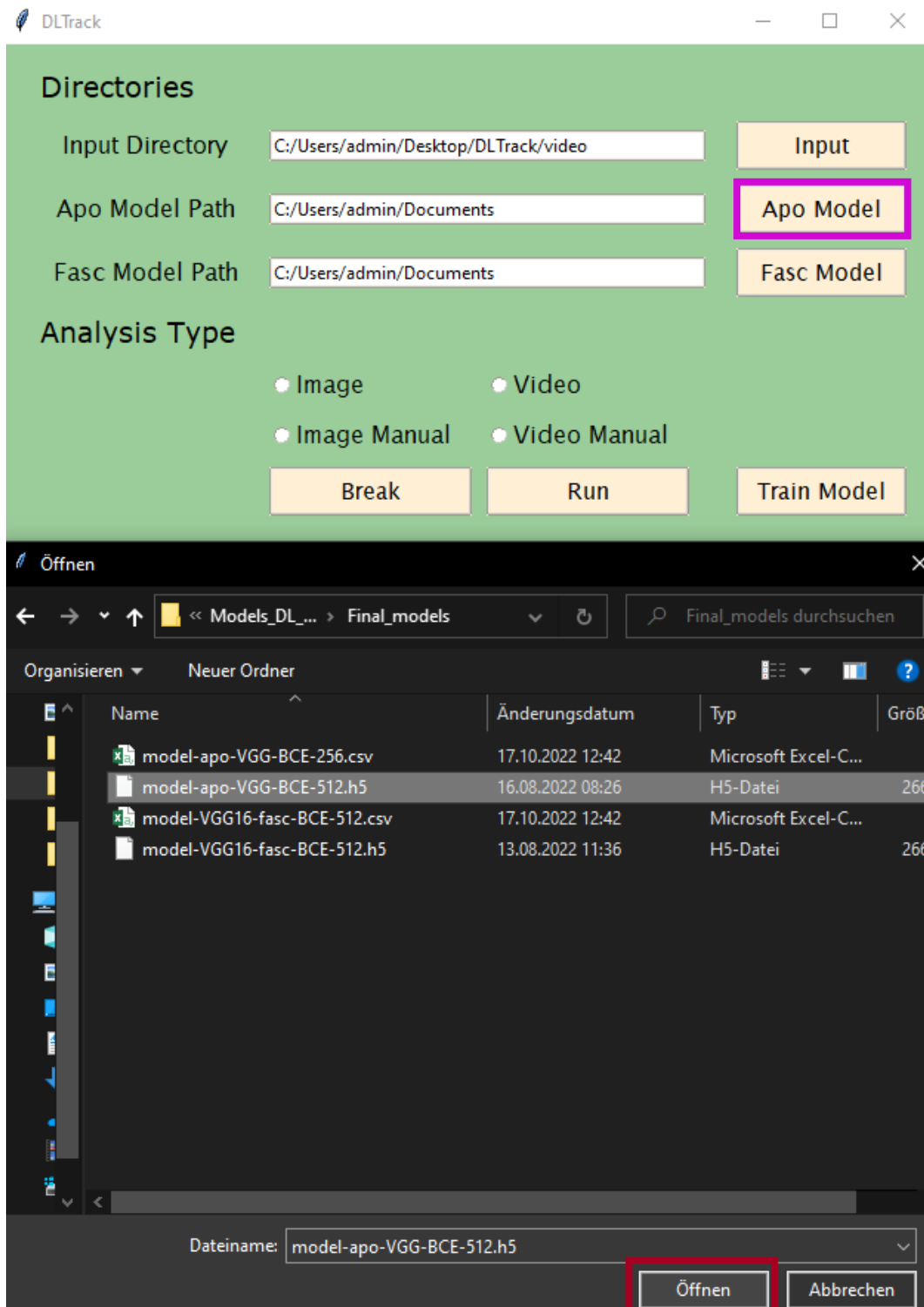
## 2. Specifying Input Directories in the GUI

Finally you can start with the actual analysis! The first step of every analysis type in DL\_Track is to specify the input directories in the graphical user interface (GUI). We assume that you have already opened the GUI. (If not, take a look at the first chapter of this document “Starting the GUI”).

You will begin with specifying the path to the folder containing the **video** to be analysed. Remember this was the folder “DL\_Track\_example/**video**”. By clicking on the **Input** button in the GUI a selection window opens where you need to select the images folder. Click **select folder** to specify the path in the GUI.

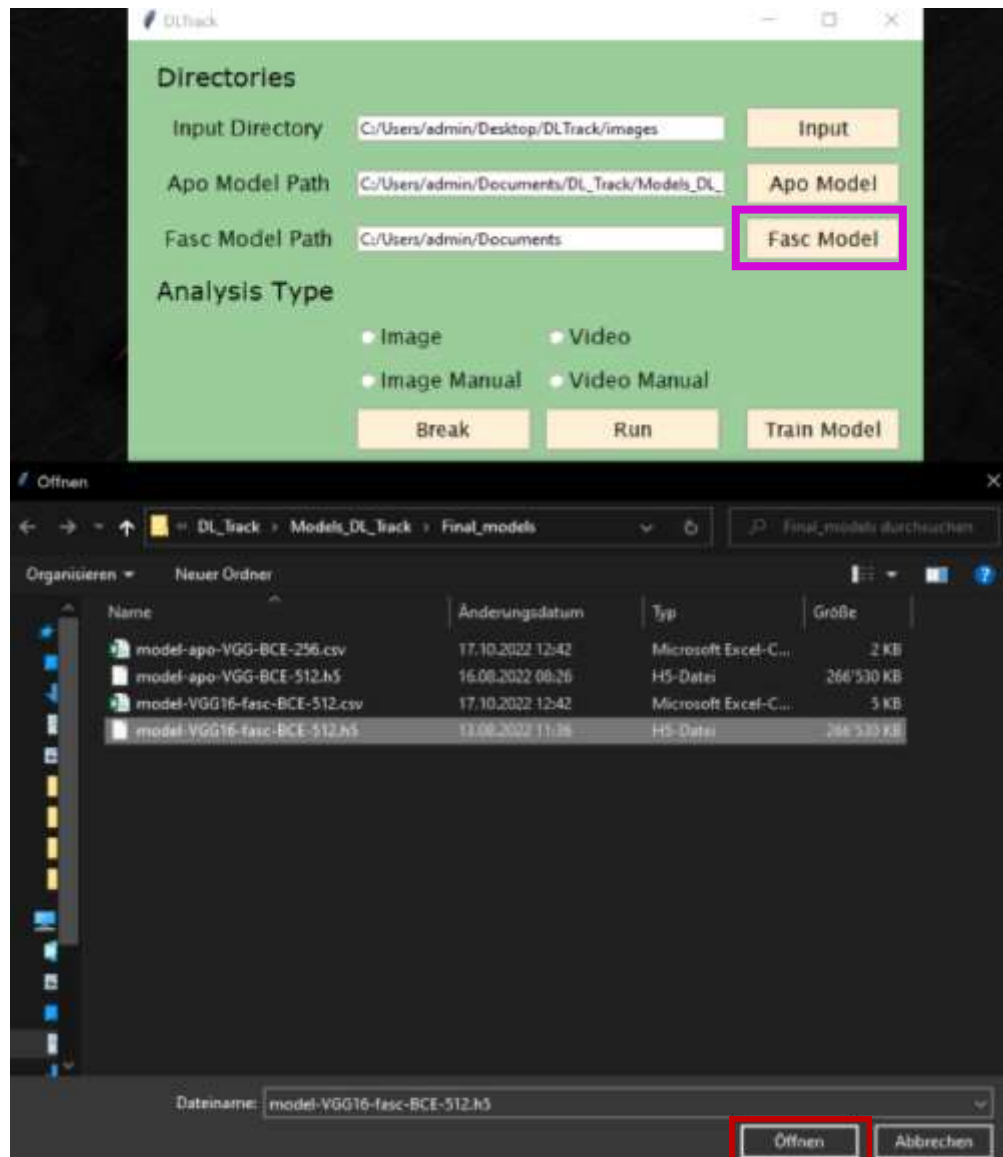


Now, you will specify the absolute path to the **aponeurosis neural network**. Remember that you placed it in the “DL\_Track\_example/**models**”. By clicking on the **Apo Model** button in the GUI a selection window opens where you need to select the **aponeurosis neural network** in the models folder. Click **open** to specify the path to the **aponeurosis neural network** in the GUI.





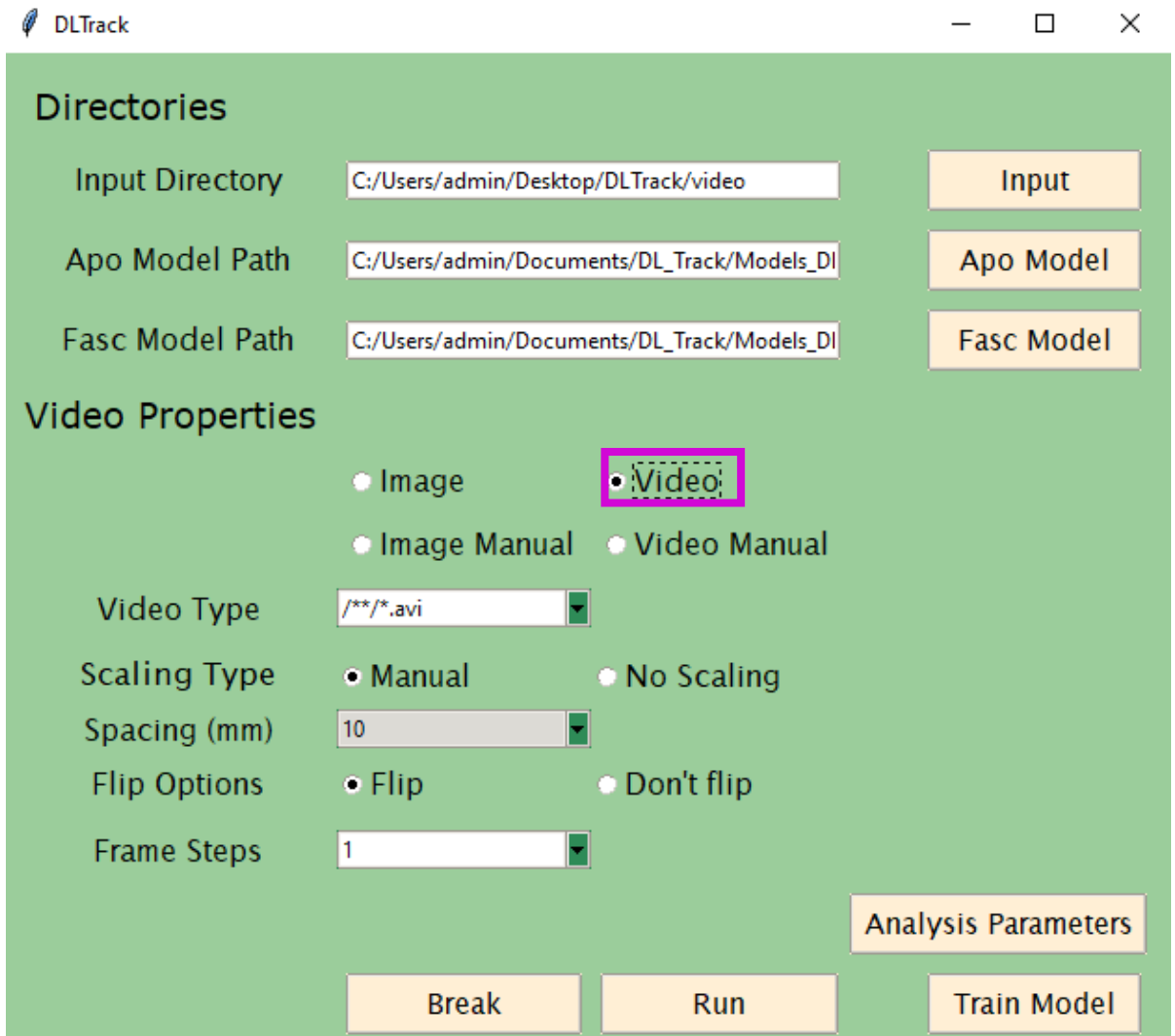
Next, you will specify the absolute path to the **fascicle neural network**. Remember that you placed it in the “DL\_Track\_example/**models**”. By clicking on the **Fasc Model** button in the GUI a selection window opens where you need to select the **fascicle neural network** in the models folder. Click **open** to specify the path to the **fascicle neural network** in the GUI.



You have now successfully defined all the input directories required for automated image analyses with DL\_Track. In the next section you will specify all relevant analysis parameters, including the analysis type. We will also explain what each parameter is used for.

### 3. Specifying Relevant Parameters

As a first step, you will select the right analysis type in the GUI. Since this section is about automated video analysis, please select the **Video** radiobutton. You can see that the GUI unfolds and several other parameters appear. You will set those in the next steps on the next page.



The screenshot shows the DLTrack application window. The 'Directories' section contains three rows: 'Input Directory' with the path 'C:/Users/admin/Desktop/DLTrack/video' and an 'Input' button; 'Apo Model Path' with the path 'C:/Users/admin/Documents/DL\_Track/Models\_DI' and an 'Apo Model' button; and 'Fasc Model Path' with the path 'C:/Users/admin/Documents/DL\_Track/Models\_DI' and a 'Fasc Model' button. The 'Video Properties' section has three radio buttons: 'Image', 'Video' (which is selected and highlighted with a red dashed box), and 'Image Manual'. Below these are 'Video Manual' and 'Video Type' (a dropdown menu showing '/\*/\*.\*.avi'). Further down are 'Scaling Type' with 'Manual' and 'No Scaling' radio buttons, 'Spacing (mm)' with a dropdown menu showing '10', 'Flip Options' with 'Flip' and 'Don't flip' radio buttons, and 'Frame Steps' with a dropdown menu showing '1'. At the bottom right is an 'Analysis Parameters' button. At the very bottom are three buttons: 'Break', 'Run', and 'Train Model'.

DLTrack

Directories

Input Directory

Apo Model Path

Fasc Model Path

Video Properties

☐ Image ☒ Video ☐ Image Manual ☐ Video Manual

Video Type

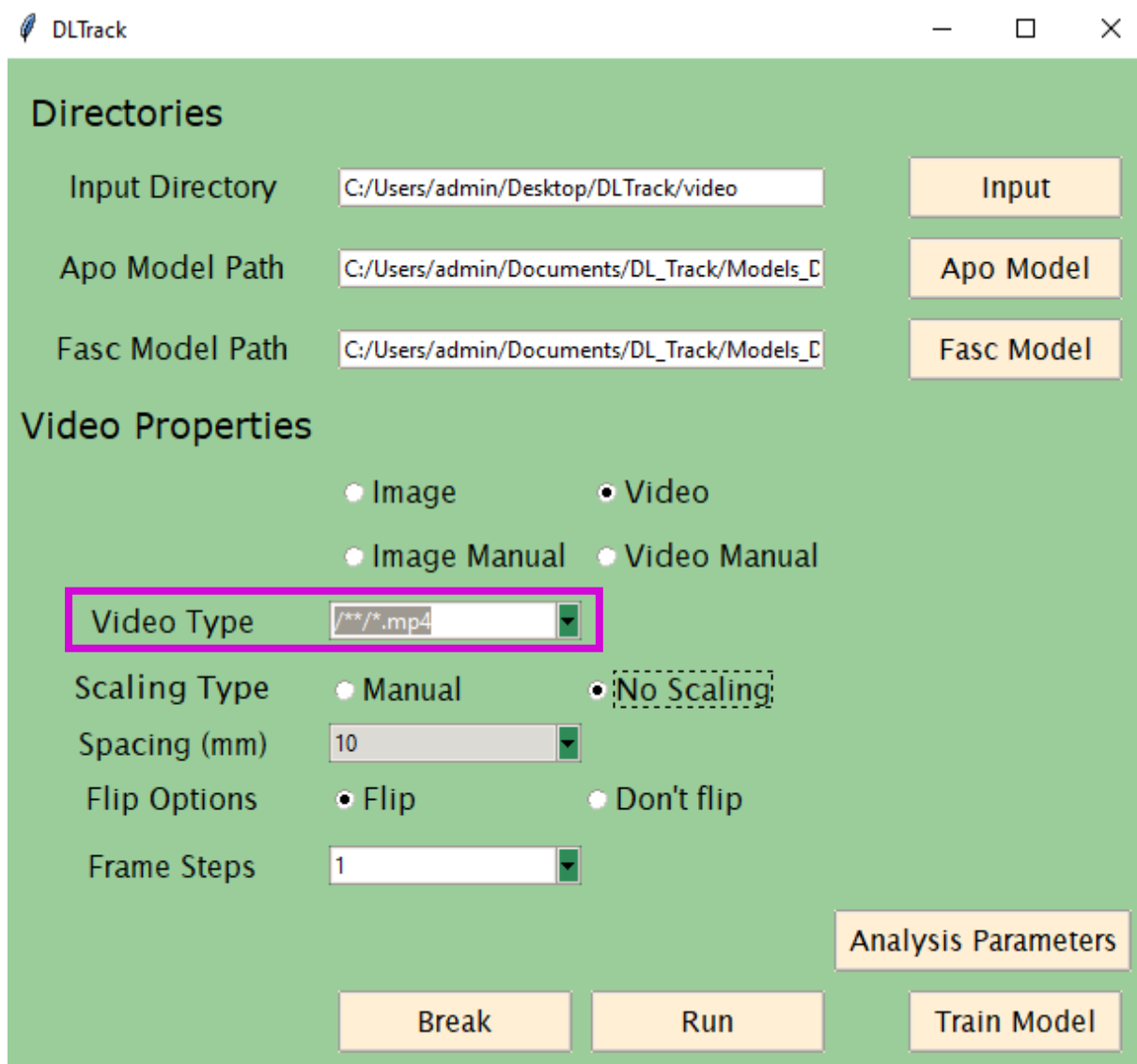
Scaling Type ☒ Manual ☐ No Scaling

Spacing (mm)

Flip Options ☒ Flip ☐ Don't flip

Frame Steps

You now need to specify the **Video Type**. The ending of the **Video Type** must match the ending of your **videos**, otherwise no files are found by DL\_Track. You can either select a pre-specified ending from the dropdown list or type your own ending. Please keep the formatting similar to those **Video Type** provided in the dropdown list. The **video** in the “DL\_Track\_example/**video**” folder are of the **Video Type** “.mp4”. Thus, you should select the “/\*\*/\*.\*.mp4” **Video Type**.



The image shows the DLTrack application window with a green background. It is divided into two main sections: "Directories" and "Video Properties".

**Directories Section:**

- Input Directory:** C:/Users/admin/Desktop/DLTrack/video (with an "Input" button to the right)
- Apo Model Path:** C:/Users/admin/Documents/DL\_Track/Models\_C (with an "Apo Model" button to the right)
- Fasc Model Path:** C:/Users/admin/Documents/DL\_Track/Models\_C (with a "Fasc Model" button to the right)

**Video Properties Section:**

- Video Type:** A dropdown menu showing "/\*\*/\*.\*.mp4". This field is highlighted with a red rectangle.
- Scaling Type:** Radio buttons for "Manual" and "No Scaling". "No Scaling" is selected and highlighted with a dashed red box.
- Spacing (mm):** A dropdown menu showing "10".
- Flip Options:** Radio buttons for "Flip" and "Don't flip". "Flip" is selected.
- Frame Steps:** A dropdown menu showing "1".

**Buttons:**

- Analysis Parameters:** A button located below the "Video Properties" section.
- Break:** A button at the bottom left.
- Run:** A button at the bottom center.
- Train Model:** A button at the bottom right.

Subsequently, you need to specify the video **Scaling Type**. Scaling in general has the huge advantage that the resulting estimated muscle architectural features are in centimetre units rather than pixel units. There are two **Scaling Types** in the DL\_Track package. For this tutorial however, you will select the “**No Scaling**” option as displayed below. We will explain the other **Scaling Type** on the next.

DLTrack

— □ ×

Directories

Input Directory

C:/Users/admin/Desktop/DLTrack/video

Input

Apo Model Path

C:/Users/admin/Documents/DL\_Track/Models\_C

Apo Model

Fasc Model Path

C:/Users/admin/Documents/DL\_Track/Models\_C

Fasc Model

Video Properties

☐ Image

☒ Video

☐ Image Manual

☐ Video Manual

Video Type

/\*\*/\*.mp4

Scaling Type

☐ Manual

☒ No Scaling

Spacing (mm)

10

Flip Options

☒ Flip

☐ Don't flip

Frame Steps

1

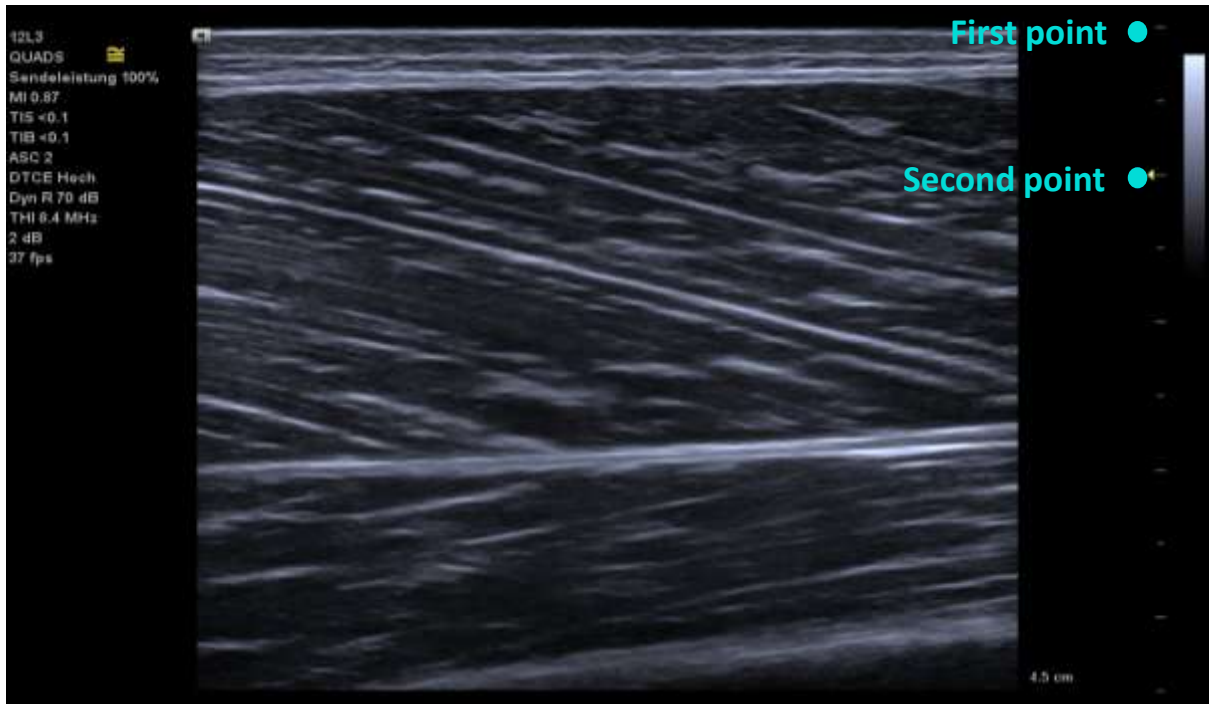
Analysis Parameters

Break

Run

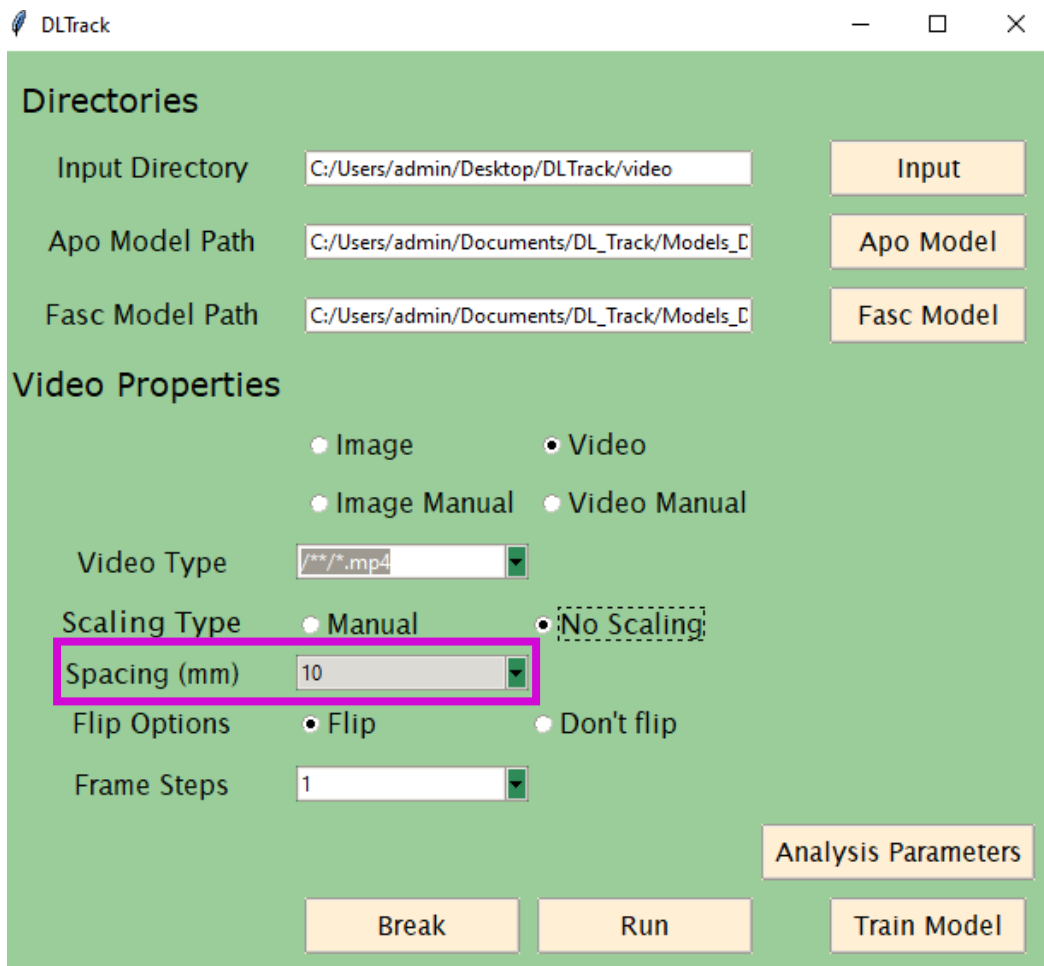
Train Model

The other **Scaling Types** is “Manual”. This **Scaling Type** requires input from the user. When you choose “Manual” as your Scaling type, you need to manually place **two points** on the first video frame using the left mouse button. This step is similar to the “Manual” scaling option for automated and manual image analysis.



No worries, you do not actually need to draw on the video frame. Just click one time with your left mouse button to record the **first point** (nothing will be displayed on the video frames during actual analysis). Place the **second point** at a known distance of either 5, 10, 15 or 20 millimetre. The distance you chose must be represented in the Scaling (see next page) parameter in the GUI.

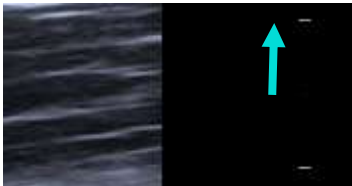
Whenever you use **“Manual”** as your Scaling Type, please make sure that the minimum distance between the scaling bars or the known distance between your manually specified points is represented in the **Spacing** parameter. For the **“No Scaling”** Scaling type, this is not necessary.



The image shows the DLTrack software interface. It has a title bar with the DLTrack logo and window controls. The main area is divided into sections: 'Directories' and 'Video Properties'. Under 'Directories', there are three rows: 'Input Directory' with a text box containing 'C:/Users/admin/Desktop/DLTrack/video' and an 'Input' button; 'Apo Model Path' with a text box containing 'C:/Users/admin/Documents/DL\_Track/Models\_D' and an 'Apo Model' button; and 'Fasc Model Path' with a text box containing 'C:/Users/admin/Documents/DL\_Track/Models\_D' and a 'Fasc Model' button. The 'Video Properties' section has radio buttons for 'Image', 'Video', 'Image Manual', and 'Video Manual'. Below these is a 'Video Type' dropdown menu showing '/\*/\*.mp4'. The 'Scaling Type' section has radio buttons for 'Manual' and 'No Scaling', with 'No Scaling' being selected and highlighted by a dashed box. Below this is a 'Spacing (mm)' dropdown menu showing '10', which is highlighted by a solid magenta box. Further down are 'Flip Options' with radio buttons for 'Flip' and 'Don't flip', and a 'Frame Steps' dropdown menu showing '1'. At the bottom right is an 'Analysis Parameters' button. At the bottom are three buttons: 'Break', 'Run', and 'Train Model'.

You can select the **Spacing** parameter only from the dropdown list as 5, 10, 15 or 20 millimetre. For this tutorial it is not necessary to select anything, as the **Spacing** parameter is not used during an analysis with Scaling Type **“No Scaling”**.

So far, we haven’t explained how to determine the minimal **distance** between the scaling bars in a video frame. This is simply the **distance** in millimeter between the two nearest scaling bars in the frame. If you do not know this **distance**, please use **“Manual”** or **“No Scaling”** Scaling Type. For example in the frame from before, the **distance** between the nearest bars is 5 millimetre. We know that because the distance between the bigger bars is always 10 millimetre.



Another parameter that you need to specify is the **Flip Options** parameters. The **Flip Options** parameter determines if the whole **video** is flipped along the vertical axis. “**Flip**” stands for flipping the video, whereas “**Don’t Flip**” means please do not flip the video. The example **video** must be flipped. Its fascicle orientation is **incorrect**, with fascicles originating at the bottom right and inserting on the top left. This would confuse our models. Below is a visual representation of a **correct** fascicle orientation. The fascicles are originating at the bottom left and are inserting on the top right. If the fascicles in your **video** outside of this tutorial are orientated differently, please specify “Flip” in the **Flip Options** parameter for those **videos**. Note that all videos in the specified input folder, in this case the DL\_Track\_example/**video**” folder, MUST have the same fascicle orientation, since the **Flip Option** is applied to all of them. It does however not matter if this is the correct or incorrect fascicle orientation.

DLTrack

Directories

Input Directory

C:/Users/admin/Desktop/DLTrack/video

Input

Apo Model Path

C:/Users/admin/Documents/DL\_Track/Models\_D

Apo Model

Fasc Model Path

C:/Users/admin/Documents/DL\_Track/Models\_D

Fasc Model

Video Properties

Image

Video

Image Manual

Video Manual

Video Type

/\*\*/\*.mp4

Scaling Type

Manual

No Scaling

Spacing (mm)

10

Flip Options

Flip

Don't flip

Frame Steps

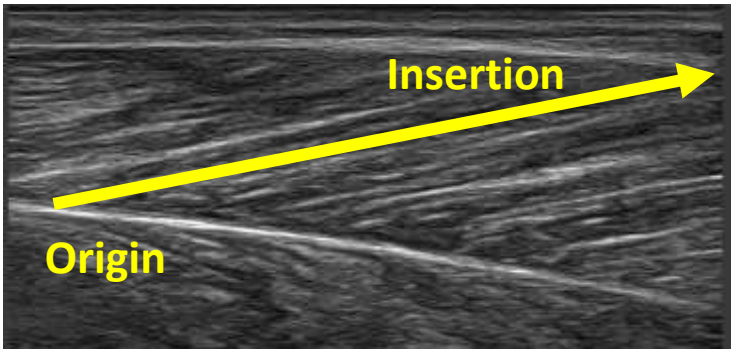
1

Analysis Parameters

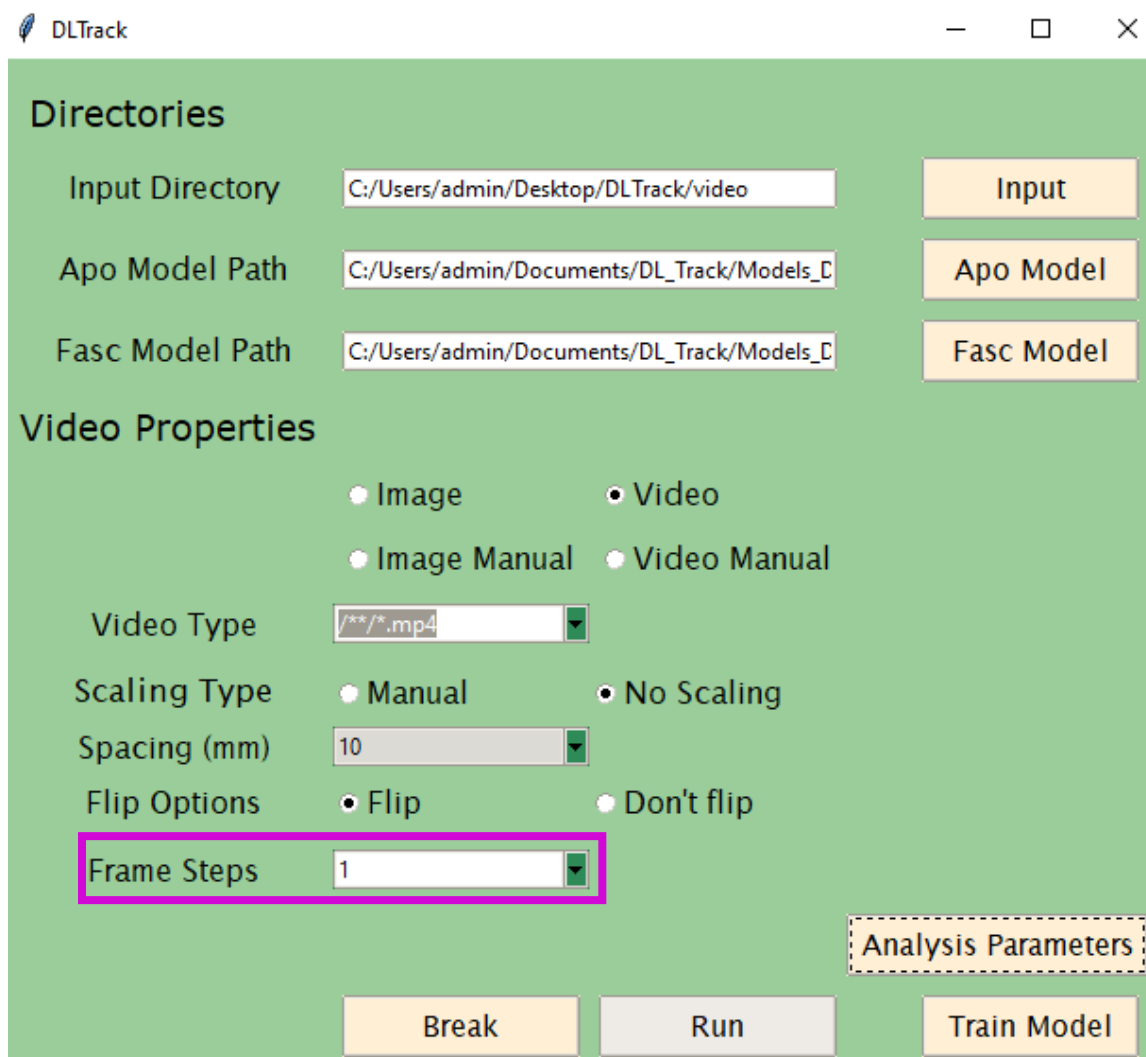
Break

Run

Train Model



The next step in the automated video analysis using the DL\_Track package is to specify the **Frame Steps**. You can either select a pre-specified **Frame Step** from the dropdown list or type your **Frame Step**. The **Frame Step** is used during the analysis as a step size while iterating through all the frames in a video. In this tutorial you should specify a **Frame Step** of 1. This means that every video frame is analysed. With a **Frame Step** of 3, every 3<sup>rd</sup> frame is analysed. With a **Frame Step** of 10, every 10<sup>th</sup> frame and so on. Although **information is lost** when you skip frames during the analysis, it also **reduces the overall analysis time**. IMPORTANT: when you skip frames in videos outside of this tutorial, the frame rate with which the video was acquired should be high enough and the captured motion a slow one. Otherwise, too much relevant information is lost.



The image shows a screenshot of the DLTrack application window. The window has a title bar with the DLTrack logo and standard window controls. The main content area is divided into two sections: "Directories" and "Video Properties".

**Directories Section:**

- Input Directory:** C:/Users/admin/Desktop/DLTrack/video (with an "Input" button to the right)
- Apo Model Path:** C:/Users/admin/Documents/DL\_Track/Models\_C (with an "Apo Model" button to the right)
- Fasc Model Path:** C:/Users/admin/Documents/DL\_Track/Models\_C (with a "Fasc Model" button to the right)

**Video Properties Section:**

- Video Type:** A dropdown menu showing "/\*\*/\*.\*.mp4".
- Scaling Type:** Radio buttons for "Manual" (selected) and "No Scaling".
- Spacing (mm):** A dropdown menu showing "10".
- Flip Options:** Radio buttons for "Flip" (selected) and "Don't flip".
- Frame Steps:** A dropdown menu showing "1", which is highlighted with a red rectangle.

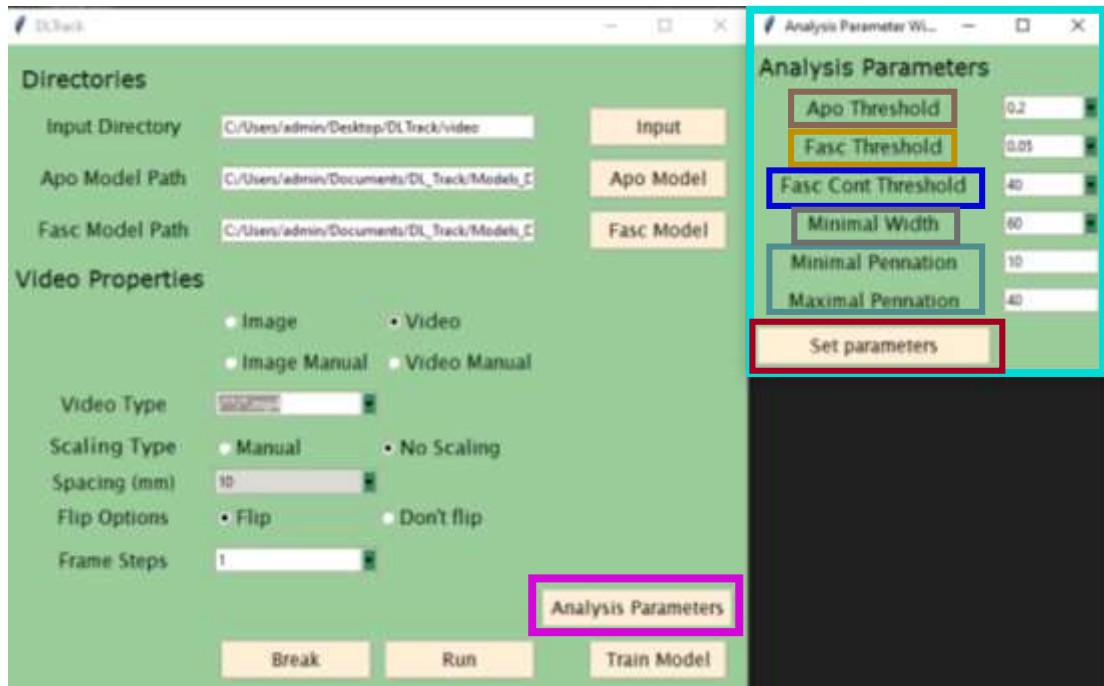
**Analysis Parameters:** A dashed box containing the text "Analysis Parameters".

**Buttons:** At the bottom, there are three buttons: "Break", "Run", and "Train Model".



## 4. Specifying Analysis Parameters

Awesome, you have now successfully selected all relevant parameters in the main GUI window. As a LAST step, you need to specify the analysis parameters for the **aponeurosis and fascicle neural networks**. When you press the **Analysis Parameters** button, a separate window will pop-up.

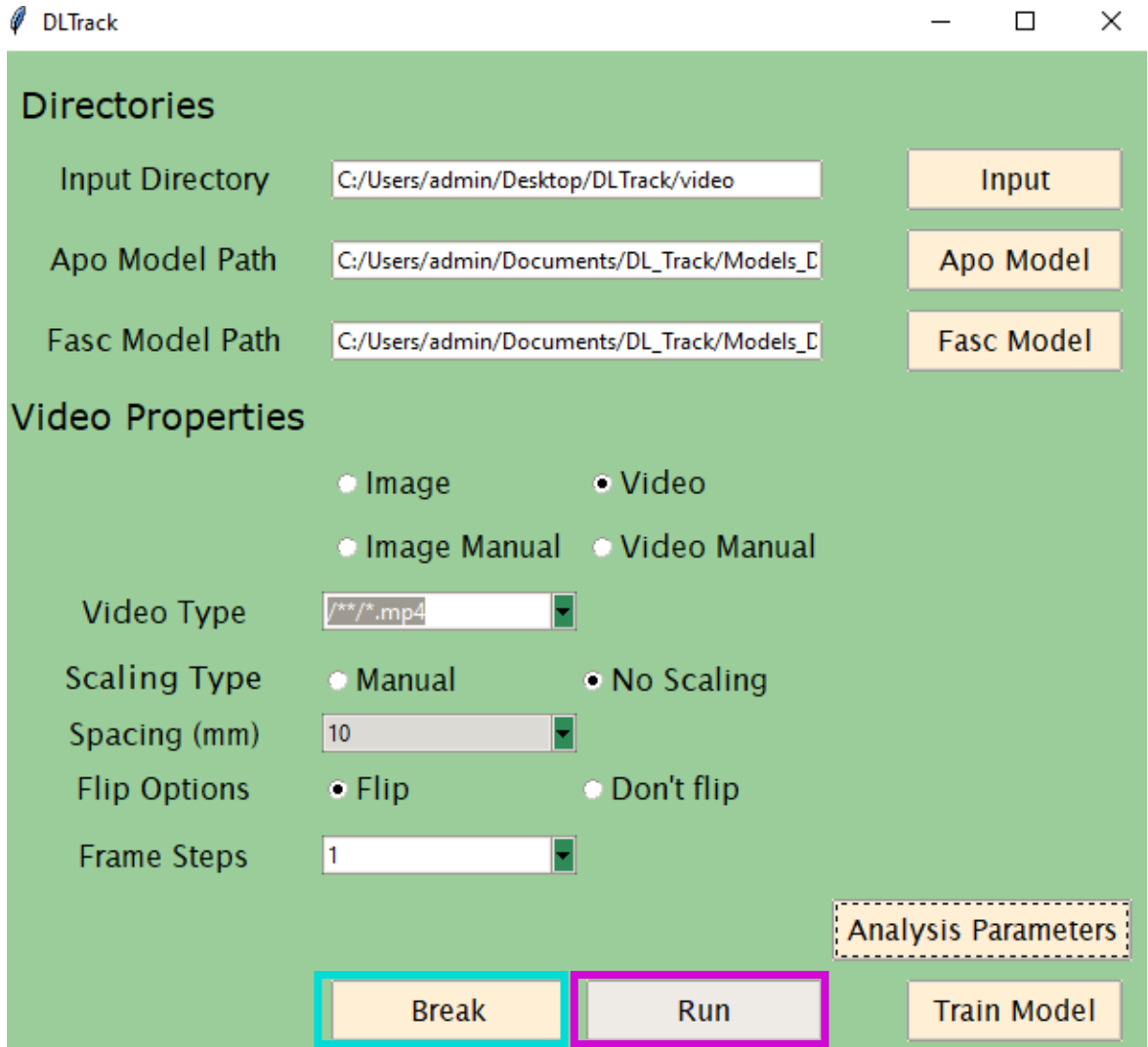


In the **Analysis Parameter window**, all parameters used by the **aponeurosis and fascicles neural networks** during inference are specified. We will give a short explanation to each of those parameters. The **Apo Threshold** parameters determines the threshold of the minimal acceptable probability by which a pixel is predicted as aponeurosis. The lower, the more pixels will be classified as aponeurosis. The **Fasc Threshold** is the same thing just for fascicle segments. The lower the **Fasc Cont Threshold**, the shorter the minimal acceptable length of detected fascicle segments to be included in the results. The **Minimal Width** determined the minimal acceptable distance between superficial and deep aponeurosis. **Minimal and Maximal Pennation** describe the respective minimal and maximal pennation angle that is physiologically possible in the analysed video frame/muscle.

For this example, you can leave all parameters the way they are. You can set the parameters by clicking the **Set parameters** button, the **Analysis Parameter window** will then close automatically. Please make sure to adapt these parameters according to your images in analyses outside of this example. For future analyses, it's best you test the ideal parameter configuration to get the best prediction results on your images in a small sample prior to the actual analysis.

## 5. Running / Breaking DL\_Track

Now its time to start the actual analysis of the example video in the “DL\_Track\_example/video” folder. You can do so by clicking the **Run** button in the main GUI window.



The image shows the DLTrack GUI window. At the top, the title bar says "DLTrack" with standard window controls. The main area is divided into two sections: "Directories" and "Video Properties".

**Directories Section:**

- Input Directory:** A text box containing "C:/Users/admin/Desktop/DLTrack/video" with an "Input" button to its right.
- Apo Model Path:** A text box containing "C:/Users/admin/Documents/DL\_Track/Models\_D" with an "Apo Model" button to its right.
- Fasc Model Path:** A text box containing "C:/Users/admin/Documents/DL\_Track/Models\_D" with a "Fasc Model" button to its right.

**Video Properties Section:**

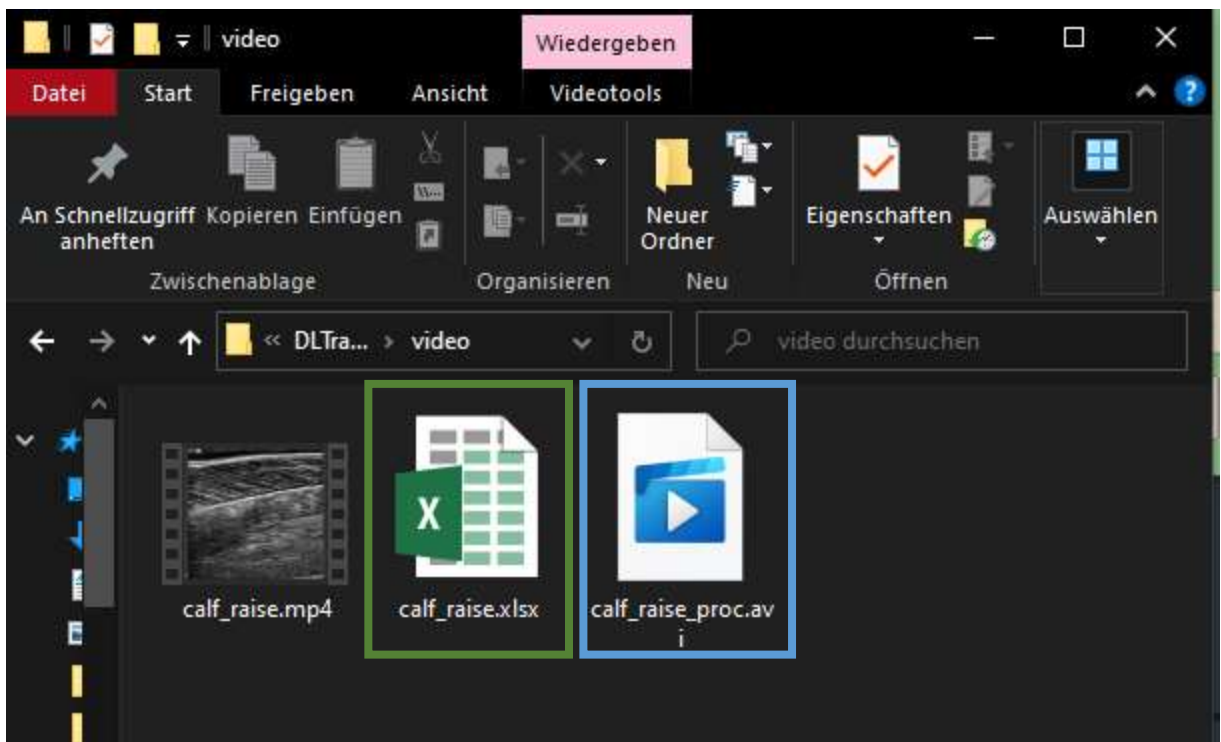
- Video Type:** A dropdown menu showing "/\*\*/\*.mp4".
- Scaling Type:** Two radio buttons: "Manual" (selected) and "No Scaling".
- Spacing (mm):** A text box containing "10".
- Flip Options:** Two radio buttons: "Flip" (selected) and "Don't flip".
- Frame Steps:** A text box containing "1".

At the bottom right, there is a dashed box labeled "Analysis Parameters". At the bottom center, there are three buttons: "Break" (highlighted with a red border), "Run" (highlighted with a green border), and "Train Model".

When you started the GUI using the command prompt in any way, you will see that the analysis is started by the statements printed in the prompt. Moreover the currently analysed frame with the segmentation results will pop up. When you started the GUI using the executable, you can see only the current frame with the segmentation results pop up. Moreover, you can see that there is a **Break** button placed in the GUI as well. Clicking the **Break** button allows you to stop the analysis at any point. The currently evaluated frame will be processed and then the analysis is terminated.

Take a look at the next pages to see what happens during the analysis and once the analysis is finished.

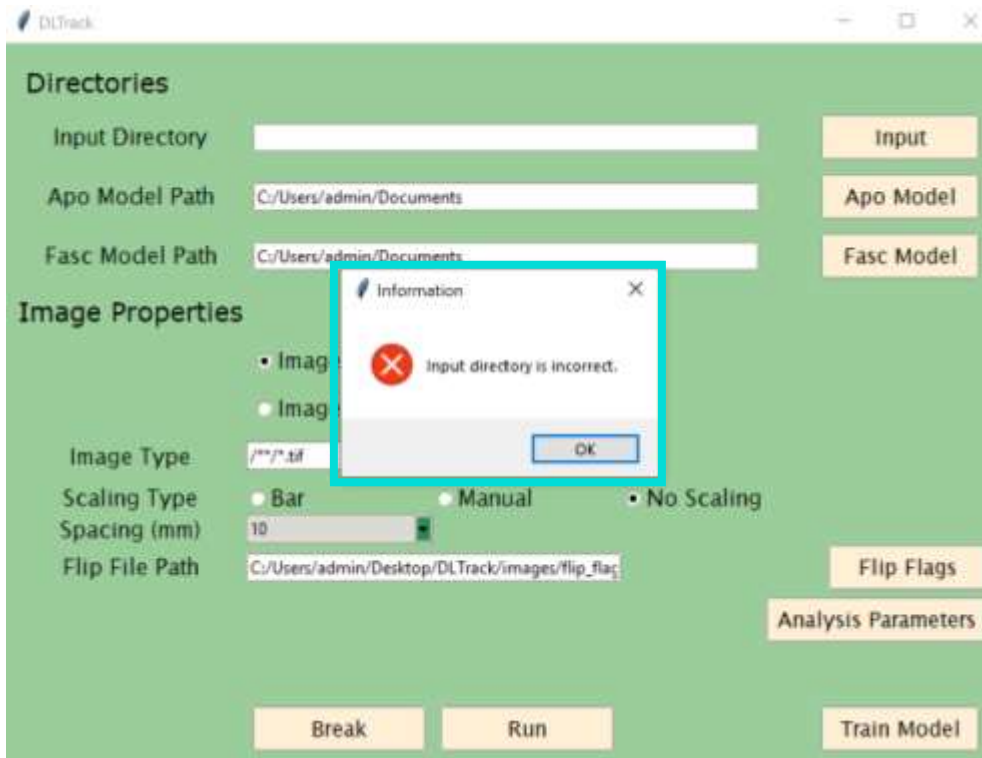
Subsequently to clicking the Run button in the main GUI, navigate again to the “DL\_Track\_example/video” folder in your explorer. You will see that two files will be / have been created, **calf\_raise\_proc.avi** and **calf\_raise.xlsx**. The **calf\_raise\_proc.avi** file contains each the input video with overlaid segmented fascicles and aponeurosis. This file allows you to visually inspect the model outputs. In your future analysis outside of this tutorial, you should always visually inspect the **calf\_raise\_proc.avi** file. The **calf\_raise.xlsx** file contains the actual architectural parameter estimates for each video frame. There, all detected muscle fascicle lengths and pennation angles as well as the calculated muscle thickness will be displayed. Each video frame is displayed in a separate row. Note that the **calf\_raise\_proc.avi** file can be opened only after the **calf\_raise.xlsx** was created.



When both files can be opened and you can see the analysis results, original image and the prediction result, we must congratulate you! You have now officially and successfully completed the DL\_Track tutorial for automated video analysis! There is one more thing though, error handling. Take a look at the next section to get more information.

## 6. Error handling

Whenever an error occurs during the analysis process, the DL\_Track GUI will open a **messagebox**. This looks always similar to this:



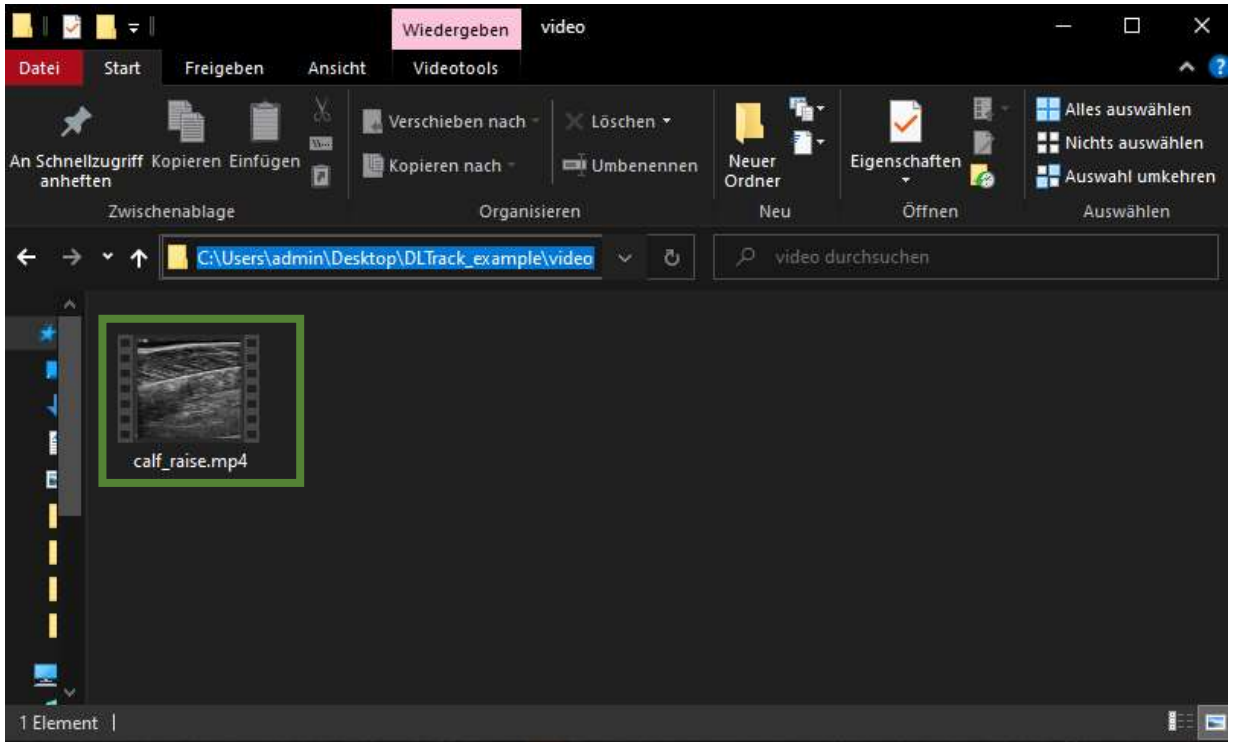
We tried to formulate these **messageboxes** as concise as possible. Just follow their instructions to fix the error and run the analysis anew. In case an error occurs that is not caught by an error **messagebox**, don't hesitate to report this in the Q&A section in the DL\_Track discussion forum. Please take a look at the "DL\_Track\_bugreport.md" file in this folder how do best do this. Otherwise, you can contact us by email at [paul.ritsche@unibas.ch](mailto:paul.ritsche@unibas.ch), but we would prefer the other way.

# *Manual Video Analysis*

The DL\_Track python software package offers several different analysis types for analysis of human lower limb longitudinal ultrasonography images. The next and last analysis type this tutorial covers is the manual video analysis. The images are evaluated manually by drawing the muscle thickness, fascicle length and pennation angles directly on the Image. Scaling the images will ensure estimated muscle architectural parameters are converted to centimetre units. For this type of analysis, single videos are a prerequisite. These videos should be contained in a single folder, like in the “DL\_Track\_examples/videos\_manual” folder. If you haven’t downloaded this folder, please do so now (link: [https://github.com/DFG-Lab/DL\\_Track/blob/master/examples/videos\\_manual](#)). Unzip the folder and put it somewhere accessible, for example on your desktop. We will make use of the included example files extensively during this tutorial. The manual video analysis type is identical to the manual image analysis type. The only difference is that the absolute video path must be specified instead of the File Type. The video is first converted and all the contained frames are separately stored as single images. Then, each frame image is analysed separately. In the next few pages, we will look at every required step to successfully perform manual video analysis with DL\_Track.

# 1. Creating a video directory

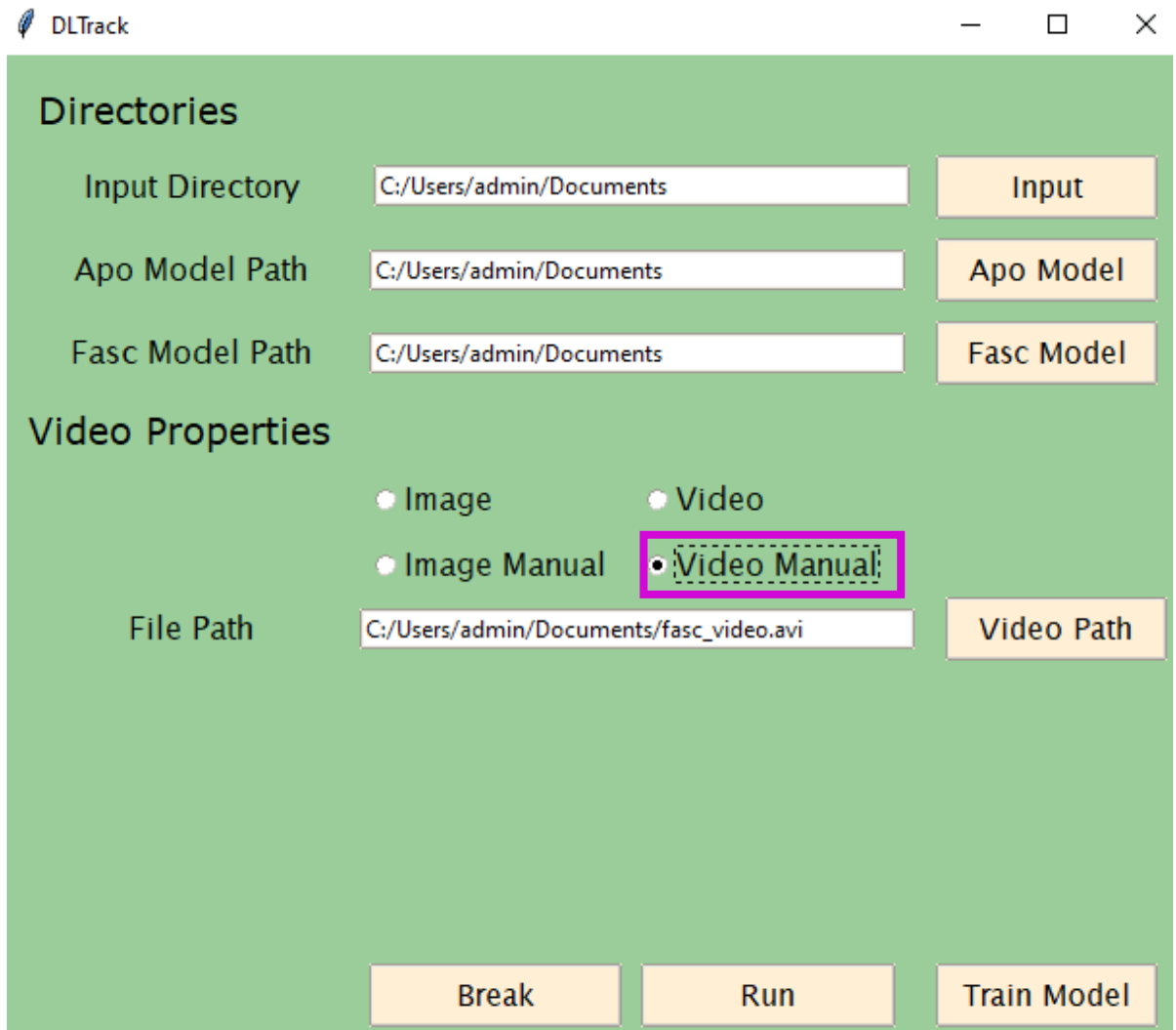
In order for DL\_Track to recognize your videos, they should best be in a single folder (though one subfolder structure is acceptable as well). Take a look how you might structure this:



You can see in the picture above that the folder contains **one video** and is **located on the desktop**. This structure is already included in the “DL\_Track\_example” folder. We will continue with demonstrating how to create folders for the **aponeurosis and fascicle neural networks** on the next page.

## 2. Specifying Relevant Parameters

For the manual video analysis DL\_Track analysis type, you do not need to specify any directories. Therefore, as a first step already, you will select the right analysis type in the GUI. Since this section is about manual video analysis, please select the **Video Manual** radiobutton. You can see that the GUI unfolds and another parameter appears. You will set this one in the next step on the next page.



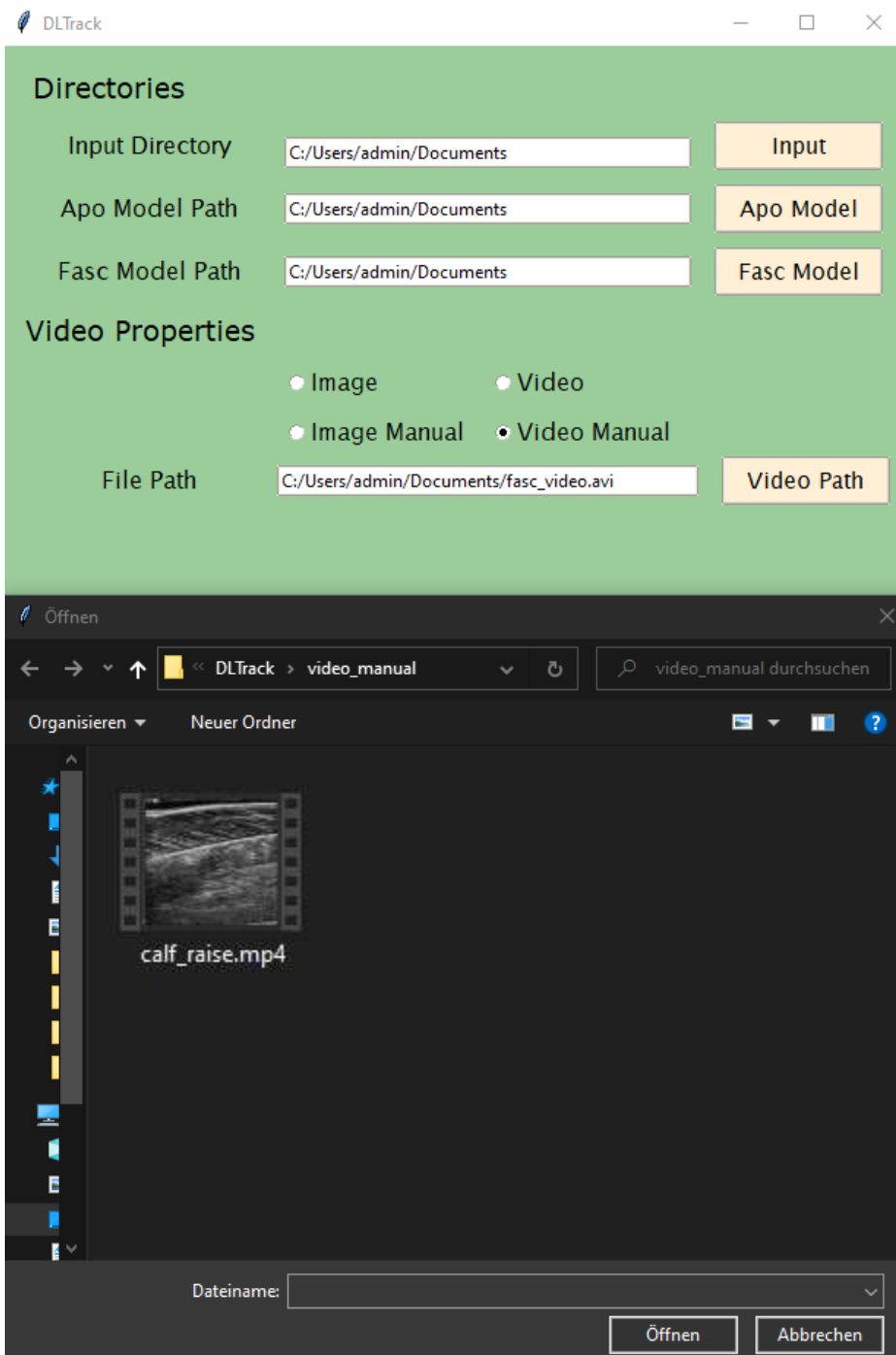
The screenshot shows the DLTrack application window. It has a title bar with the icon and text 'DLTrack', and standard window controls (minimize, maximize, close). The main area is divided into two sections: 'Directories' and 'Video Properties'. In the 'Directories' section, there are three rows, each with a label, a text input field, and a button. All three input fields contain the path 'C:/Users/admin/Documents'. The buttons are labeled 'Input', 'Apo Model', and 'Fasc Model'. The 'Video Properties' section contains a group of radio buttons for selecting the analysis type: 'Image', 'Video', 'Image Manual', and 'Video Manual'. The 'Video Manual' option is selected and highlighted with a red dashed border. Below the radio buttons is a 'File Path' label, a text input field containing 'C:/Users/admin/Documents/fasc\_video.avi', and a 'Video Path' button. At the bottom of the window, there are three buttons: 'Break', 'Run', and 'Train Model'.

Directories		
Input Directory	<input type="text" value="C:/Users/admin/Documents"/>	<input type="button" value="Input"/>
Apo Model Path	<input type="text" value="C:/Users/admin/Documents"/>	<input type="button" value="Apo Model"/>
Fasc Model Path	<input type="text" value="C:/Users/admin/Documents"/>	<input type="button" value="Fasc Model"/>

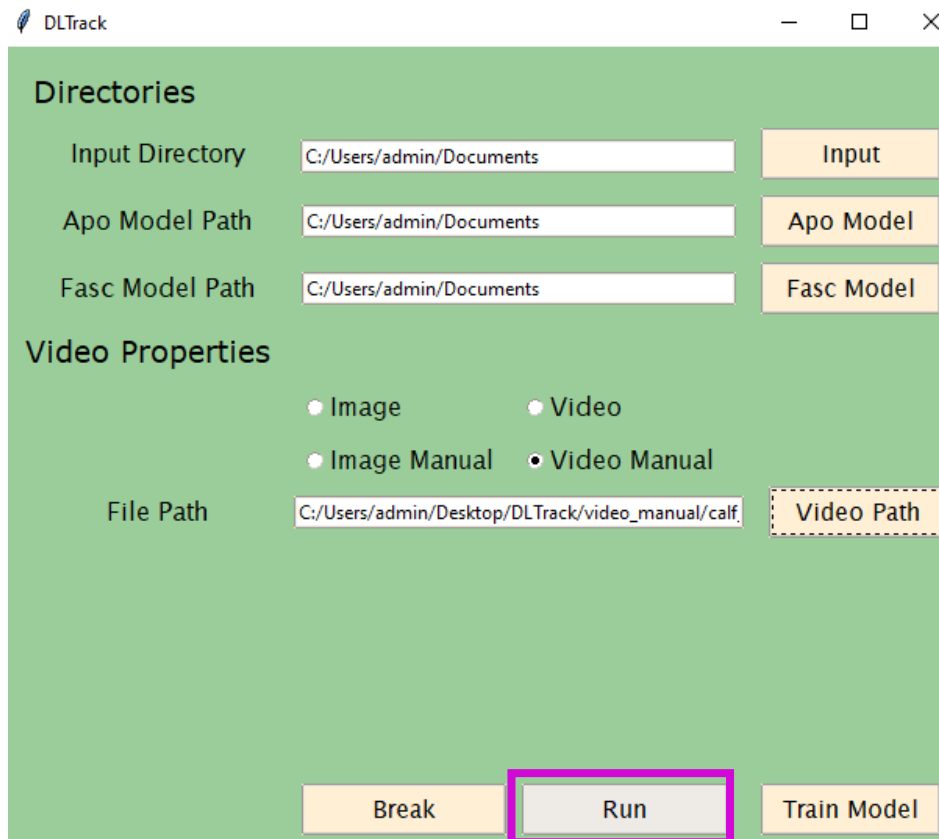
Video Properties	
	<input type="radio"/> Image <input type="radio"/> Video
	<input type="radio"/> Image Manual <input checked="" type="radio"/> Video Manual
File Path	<input type="text" value="C:/Users/admin/Documents/fasc_video.avi"/> <input type="button" value="Video Path"/>

Next, you need to specify the absolute **File Path** of the **video file** to be analysed. Remember that the example **video file** is placed in the “DL\_Track\_example/**video\_manual**” folder. By clicking on the **Video Path** button in the GUI, a selection window opens where you need to select the example **video file** in the **video\_manual**. Click **open** to specify the path to the **video file** in the GUI.





Allright, once you have specified the video file path, you can start with the analysis of the **example video** contained in the “DL\_Track\_example/**video\_manual**” folder. You can start the analysis by clicking the **Run** button in the main GUI.



Once you clicked the Run button, the “Manual Analysis window” will pop up. If this happens, congrats to you! You have entered all relevant parameters for the manual video analysis correctly!

From here, all further steps are identical with the manual image analysis. The only difference though is that in the folder of the inputted video, a new folder is created containing all the single image frames.

The scaling of the image, extending of the aponeuroses, single segment muscle thickness measurements, three segment muscle fascicle measurement and two segment pennation angle measurement are identical. Saving the results (with the very important button), continuing to the next image frame, terminating the analysis process and error handling is identical. **Therefore, we kindly refer you to section 4 of this tutorial Manual Image Analysis** (because we don’t want to repeat ourselves) **to see how the all the architectural parameters are analysed.** We are confident that you can do this on your own now!

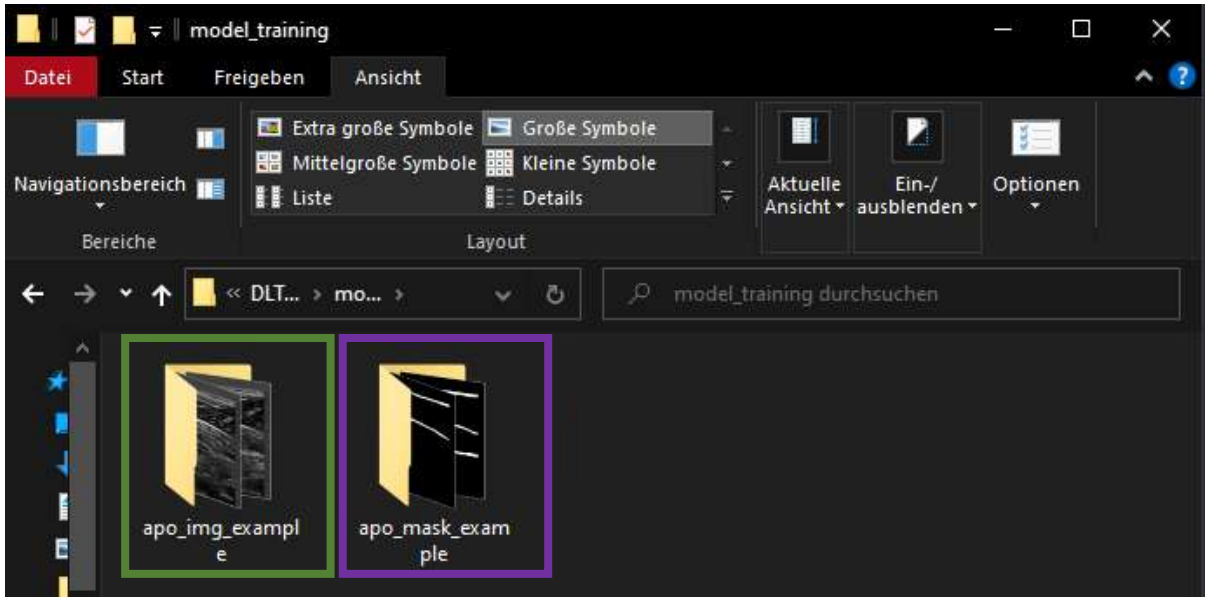
# Training Your Own Networks

Not only does the DL\_Track python software package offer four different analysis types for musculoskeletal ultrasonography images and videos of human lower limb muscles. The package also includes the possibility to train your own neural networks that may be better suited for images with different characteristics than those in our training data. This is also embedded in the GUI. It is advantageous to have a working GPU setup, otherwise model training will take much longer. How to setup your GUI for DL\_Track is described in the installation guidelines of our [Github repository](#). In the next few pages, you will learn to train your own neural networks. This is also the last part of this tutorial. After completion of this chapter, you will know the DL\_Track GUI as well as the back of your hand. If you don't have any experience with training deep neural networks, we strongly advise to work with the pre-defined settings. Otherwise, you are of course free to choose. For this introductory tutorial, we would however like you to use the pre-defined settings no matter what your experience level is. Although you can generally adapt a number of parameters during training, you cannot change the neural network architecture from the GUI (of course you could modify source code to do so). This is because during experimenting with different model architectures, we found a combination of a on imagenet<sup>1</sup> pre-trained VGG16 encoder<sup>2</sup> and a standard U-net decoder<sup>3</sup> to be the best performing model. Thus, all the models trained using the GUI will have this architecture. To explain you the parameters used during model that are adaptable from the GUI is out of the scope of this tutorial. However, we would like to refer you to [this excellent introductory course](#) in case you are a deep learning beginner. Training your own networks for muscle architecture analysis requires pairs of original images and manually labelled masks. Examples are provided for you in the "DL\_Track\_example/model\_training" folder. If you haven't downloaded this folder, please do so now (link: ). Unzip the folder and put it somewhere accessible, for example on your desktop. We will make use of the included example files extensively during this tutorial. In this tutorial, you will only learn to train model that segment the muscle aponeuroses. Yes that's right, segment the fascicles requires a separate model because the task is so different. However, everything works the same, you just need to use fascicle images and masks instead of aponeurosis images and masks. Examples of those are provided in the "DL\_Track\_example/model\_training" folder as well.

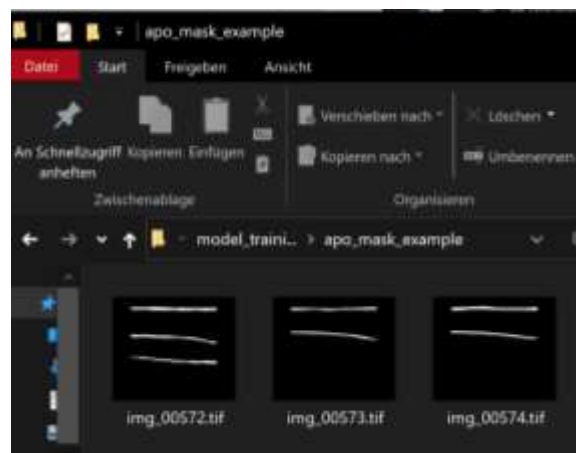
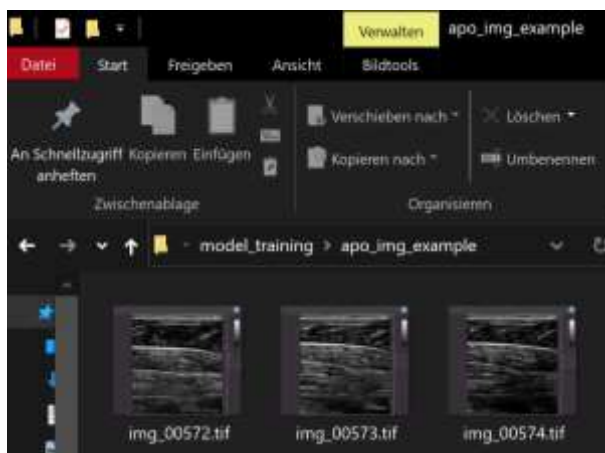
Anyway, enough said. Let's get to the model training part. The most important part is data preparation and labelling. This is where you will start.

# 1. Data preparation and image labeling

In order for DL\_Track to recognize your **aponeurosis images** and **aponeurosis masks**, they should best be in different single folders. Take a look how you might structure this:



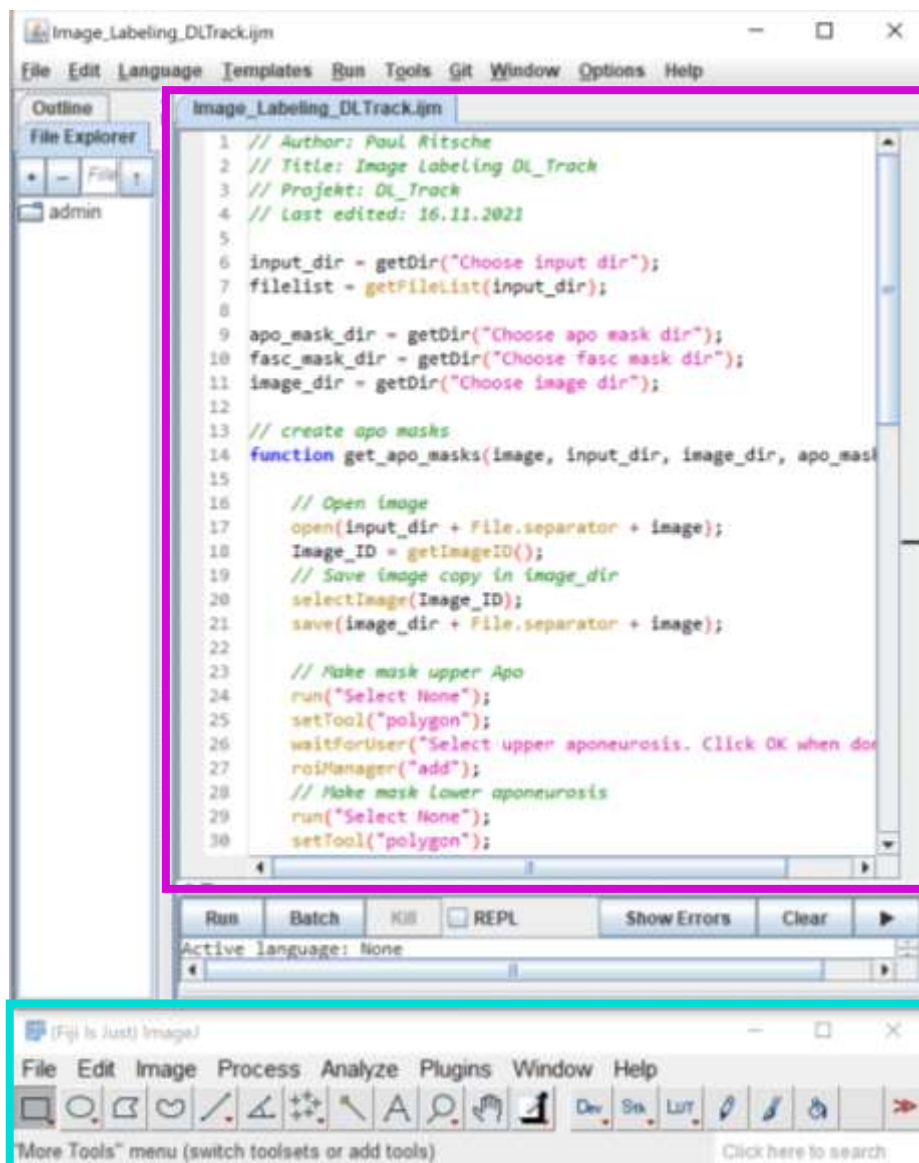
You can see in the picture above that the “DL\_Track\_example/model\_training” folder contains two subfolders, “**apo\_img\_example**” and “**apo\_mask\_example**”. The original images are located in the “**apo\_img\_example**” folder whereas the corresponding masks are located in the “**apo\_maks\_example**” folder. We advise you to keep a similar folder structure when you train your own models outside of this tutorial. When you take a look below, you can see that the **original image** and the **corresponding masks** have exactly the same name. This is **SUPER MEGA** important. Otherwise, the model is trained using the wrong masks for the images. You do not want this!



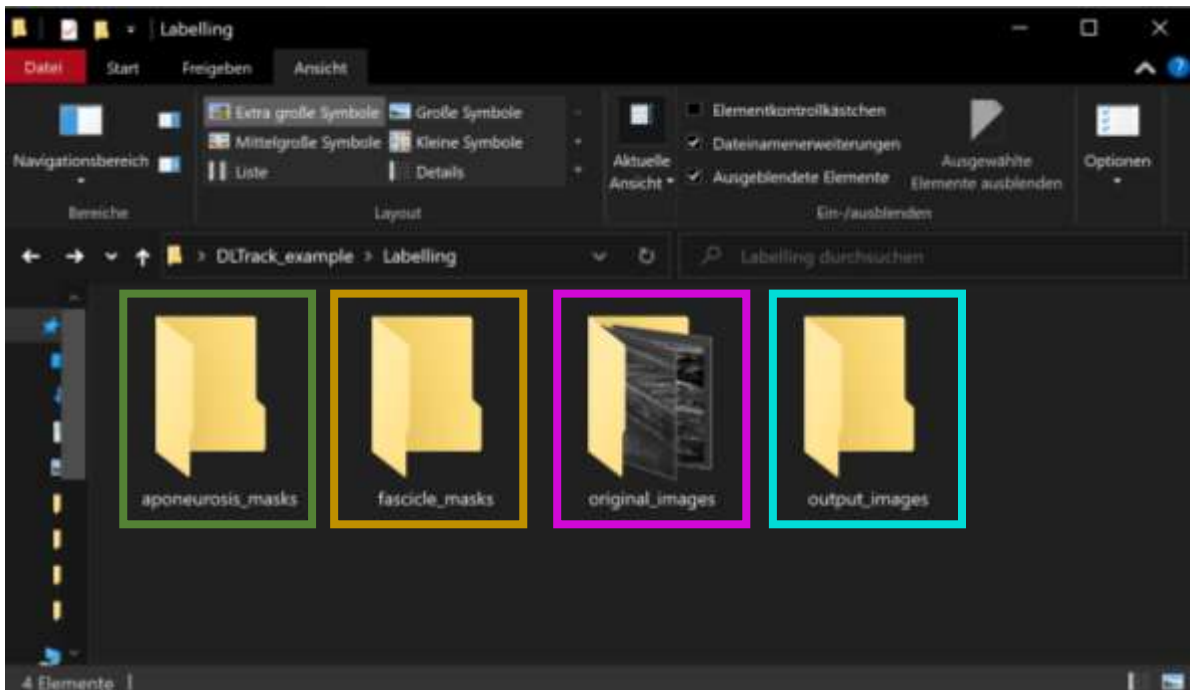
**Caution, this section is not part of the tutorial. It is merely additional information for you that is not required to run the GUI and follow the example model training tutorial. The actual tutorial continues on page 72.**

When you want to train your own networks outside of this tutorial, you need to label your original ultrasonography images of the lower limbs. We have prepared you the instructions how to use the **automated script** that we provide (this script does not automatically label the images, but automates the selection processes and image / mask saving). The software you will perform the labelling in is called **ImageJ / Fiji**. You can download the software [here](#). The **automated script “Image\_Labeling\_DLTrack.ijm”** is located in the folder “DL\_Track/docs/labeling/” in our [Github repository](#).

The easiest way to run the **“Image\_Labeling\_DLTrack.ijm”** script is by simply drag and drop it in the **Fiji / ImageJ** window. Therefore, **Fiji / ImageJ** must be already running. As a result, the script will be opened.

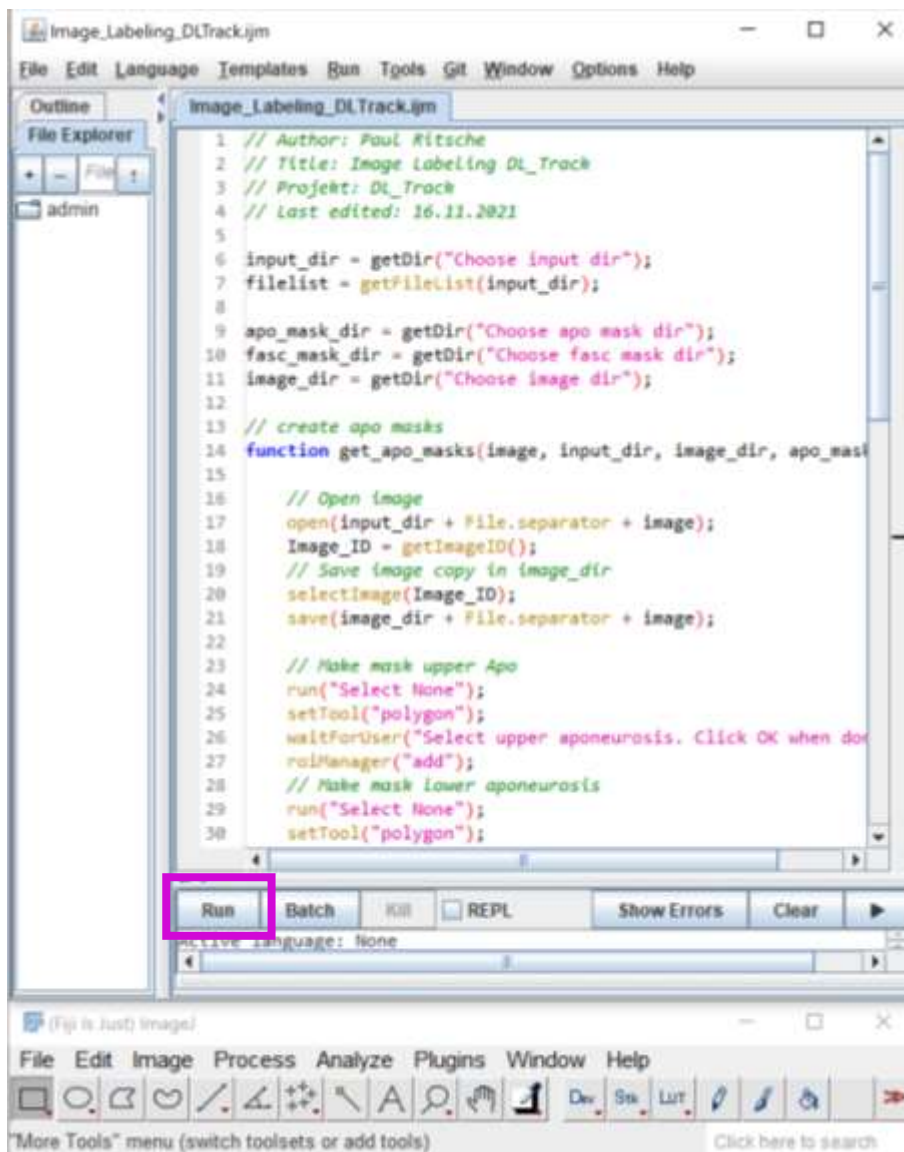


Before you can start the labelling process however, you need to create four folders in an easily accessible place (perhaps your desktop?). One folder containing the **original images** you want to label. Then create three more folders, one named **“output\_images”**, the second called **“fascicle\_masks”** and the third called **“aponeurosis\_masks”**. In the **“output\_images”** the original images are saved with an adapted name. In the **“fascicle\_masks”** and **“aponeurosis\_masks”** folder the respective masks are saved with the same name as the corresponding image in **“output\_images”**.



Take a look at the next page on how to continue once you created the above demonstrated folder structure.

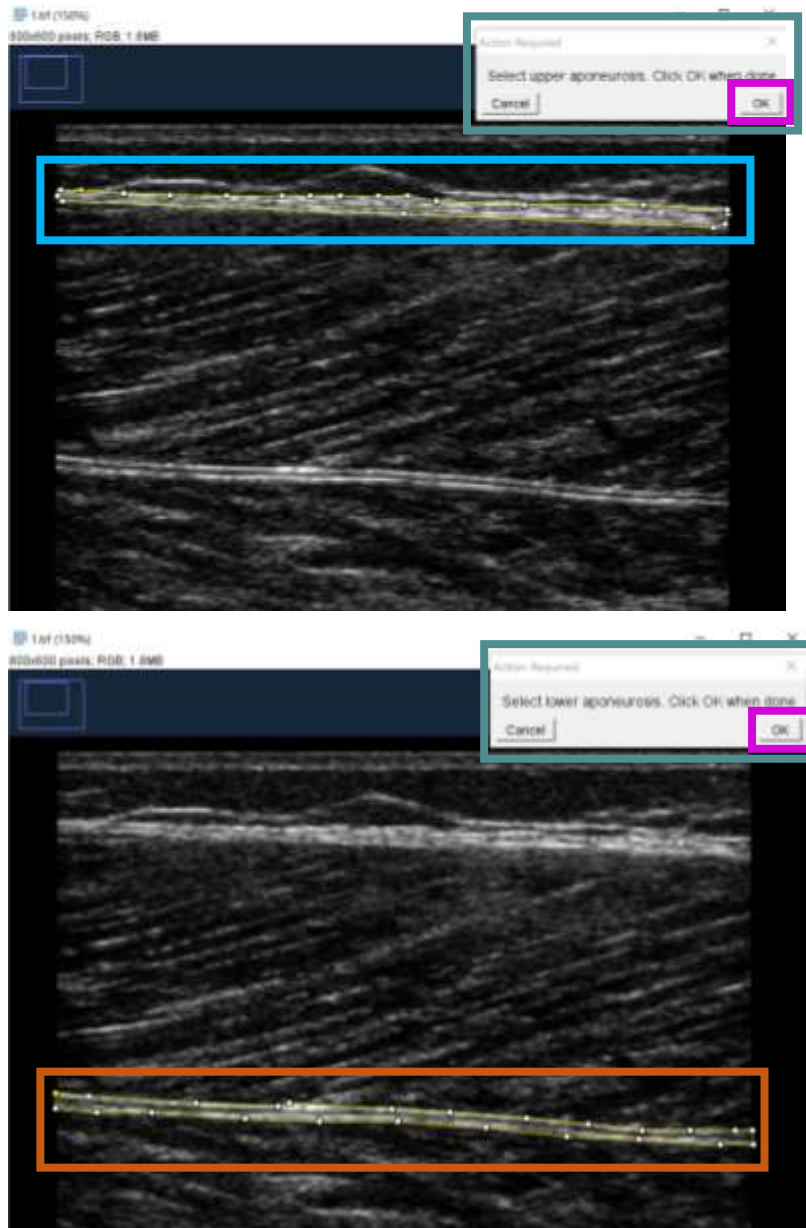
Once you created the above demonstrated folder structure, simply press the **Run** button in the Fiji / ImageJ API to start the “Image\_labelling\_DLTrack.ijm” script.



Please follow the instructions appearing in the messageboxes. To begin with, you need to specify the four directories. The first directory you need to select is the original image folder (called input dir). The second folder is the “aponeurosis\_masks” folder (called apo mask dir). The third is the “fascicle\_masks” folder (called fasc mask dir). The last folder you need to specify is the “output\_images” folder (called image dir). Subsequent to specifying the directories, you are required to create the masks. First the aponeurosis mask, then the fascicle mask. How to do this is demonstrated on the next page.

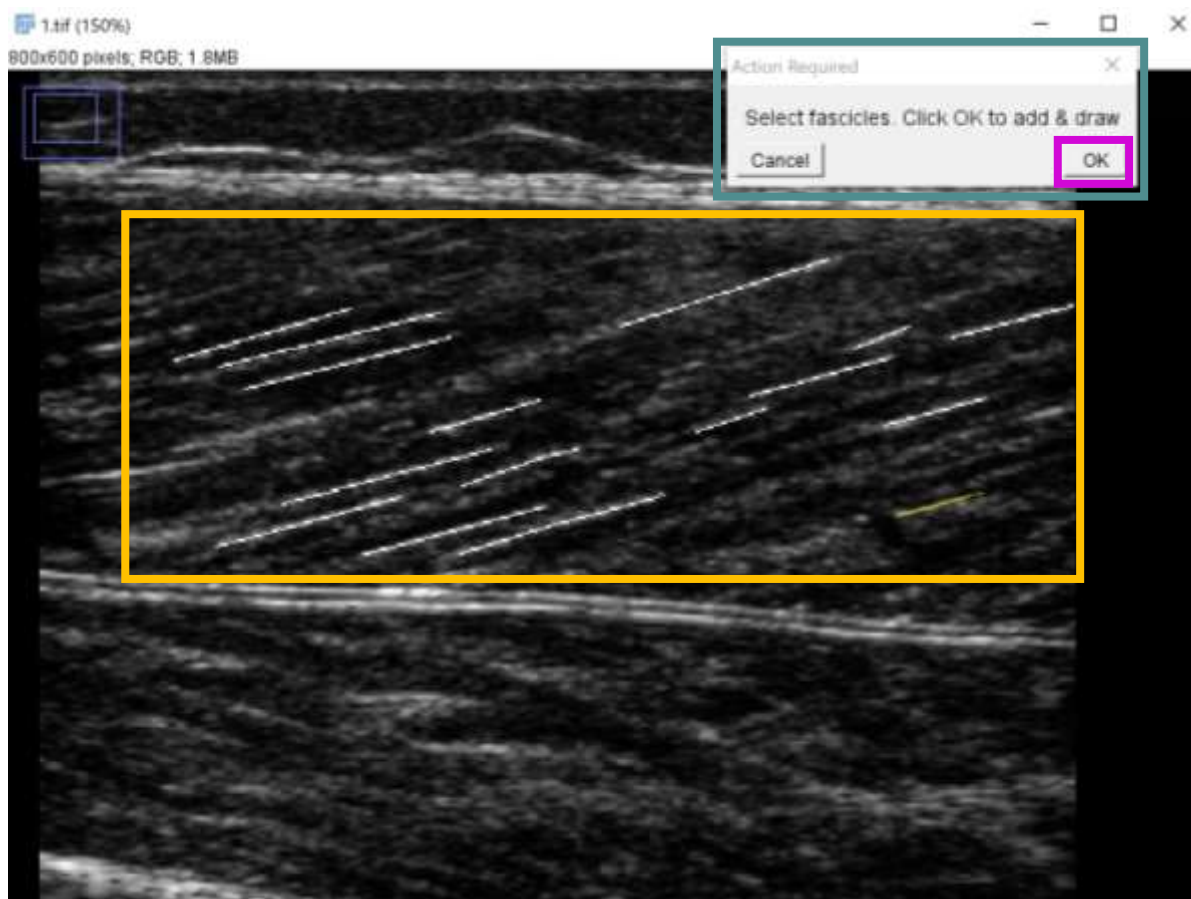


The polygon tool is selected automatically for you to draw around the **superficial aponeurosis**. Again, follow the instructions in the **messagebox**. Draw around the **superficial aponeurosis** (double click to start drawing, click to add a segment, double click to stop drawing) and once you are finished, click the **OK** button in the **messagebox** to proceed to the selection of the **lower aponeurosis**. Please be careful to only include aponeurosis tissue in your selection and no surrounding tissue. The result should look like this for the **upper** and **lower aponeurosis**:



Once you have selected the lower aponeurosis, click the **OK** button in the **messagebox** to proceed to the fascicle labelling. Take a look on the next page to see how this is done.

The segmented line tool is selected automatically for you to follow the visible fascicle segments. Again, simply follow the instructions in the **messagebox**. It is of **utmost importance** that you draw **only over the actually visible parts** of the **fascicle** segment. Make sure that you only label bright **fascicle** tissue that is clearly visible. Once you drew one **fascicle** with segmented line tool (double click to start drawing, click to add a segment, double click to stop drawing) click the **OK** button in the **messagebox** to proceed to the next **fascicle** segment. Draw as many segments as are clearly visible on the image. When you press the **OK** button in the **messagebox** without making a further selection, you will proceed to the next image in the original image folder and start again with the aponeurosis labelling. The result of your labelling should look something like this:



When you read this, you now know how to label images to train your own neural networks with your own lower limb ultrasonography images! Congrats! We will now continue with the example tutorial on how to train your own networks with the example images and masks provided.

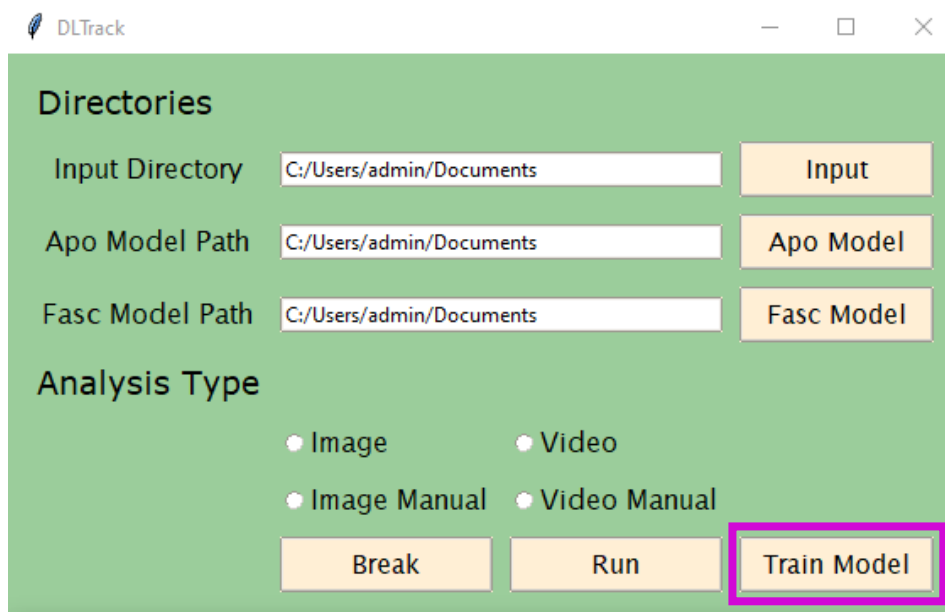


## 2. Specifying Relevant Directories

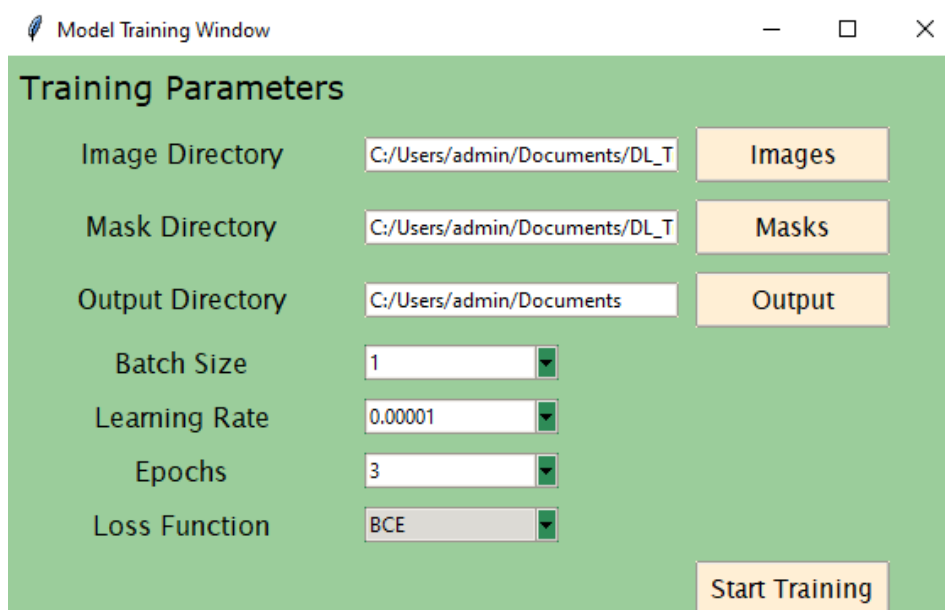
**Caution, the actual tutorial continues here!**

As a next step, you can start the GUI. If you do not know how to do this, please take a look at section two of this tutorial, Starting the GUI.

Once you started the GUI and the main GUI window opened, click on the **Train Model** button to select the relevant directories and model training parameters. The separate “Model Training window” will pop up. We will explain this window on the next page.

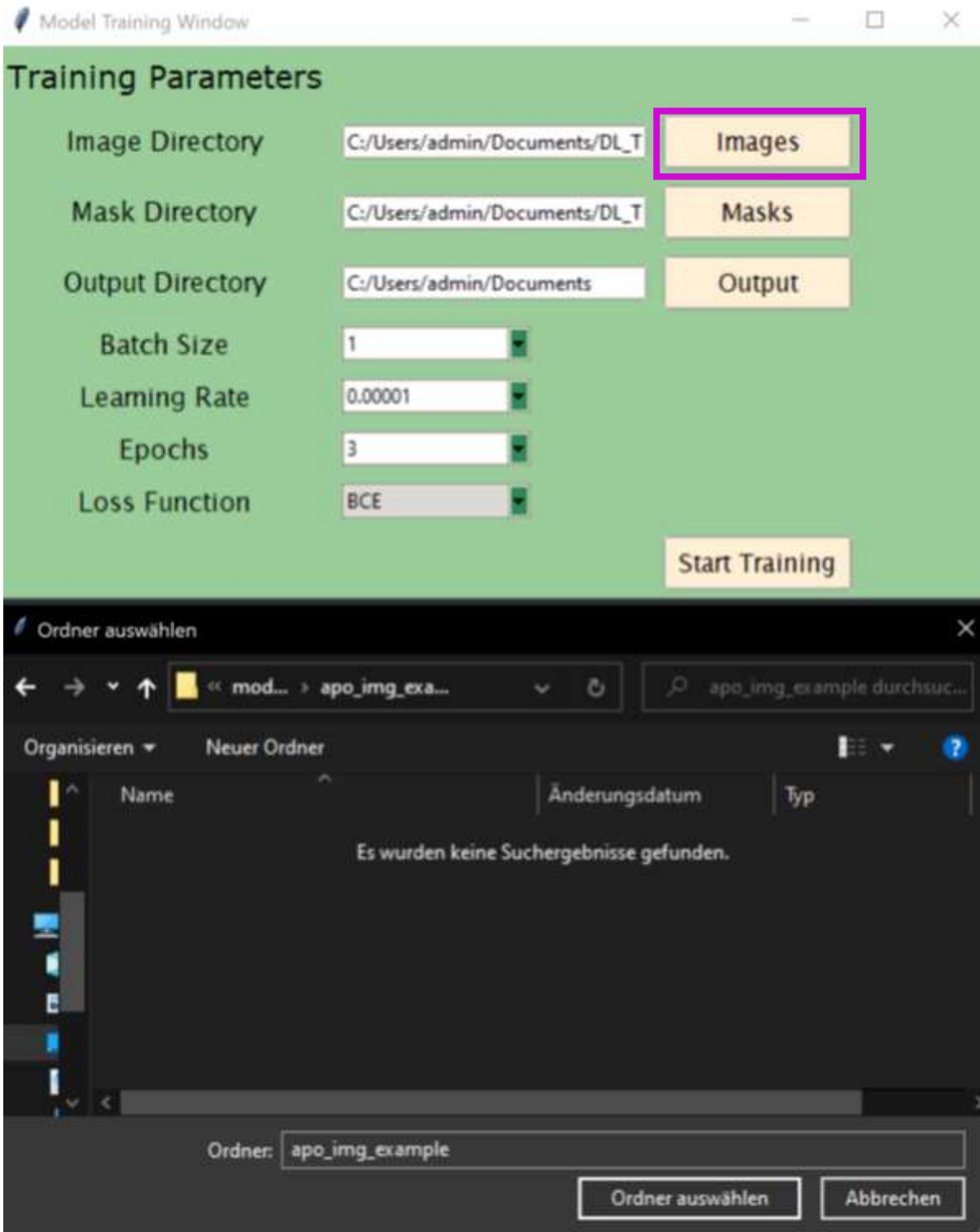


The DLTrack main GUI window has a green background. It contains three text input fields, each with a corresponding button to its right. The first row is 'Input Directory' with the path 'C:/Users/admin/Documents' and an 'Input' button. The second row is 'Apo Model Path' with the same path and an 'Apo Model' button. The third row is 'Fasc Model Path' with the same path and a 'Fasc Model' button. Below these is the 'Analysis Type' section with four radio buttons: 'Image', 'Video', 'Image Manual', and 'Video Manual'. At the bottom are three buttons: 'Break', 'Run', and 'Train Model'. The 'Train Model' button is highlighted with a red rectangular border.

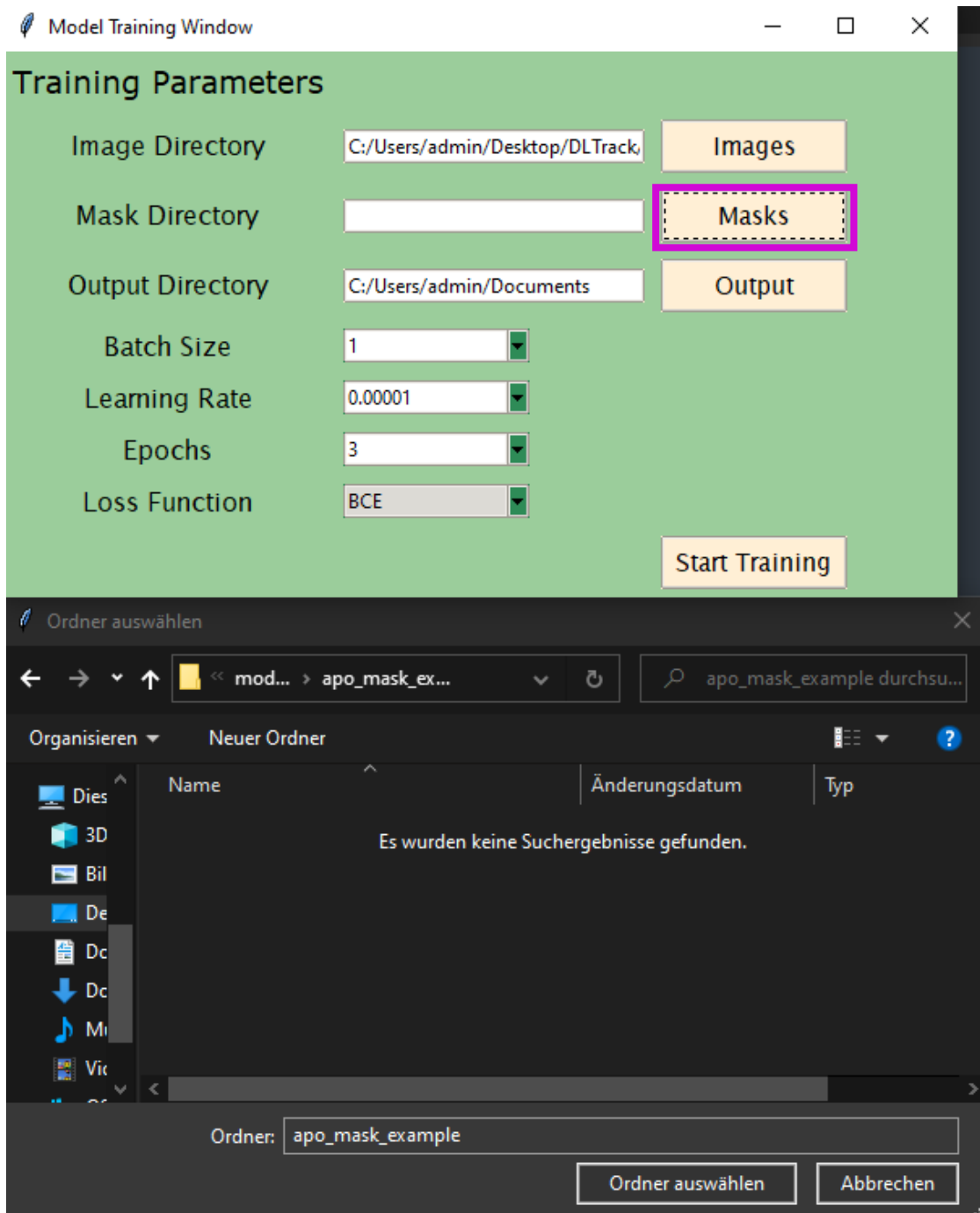


The Model Training Window also has a green background. It features six text input fields with dropdown menus to their right, and a 'Start Training' button at the bottom right. The first row is 'Image Directory' with the path 'C:/Users/admin/Documents/DL\_T' and an 'Images' button. The second row is 'Mask Directory' with the same path and a 'Masks' button. The third row is 'Output Directory' with the path 'C:/Users/admin/Documents' and an 'Output' button. The fourth row is 'Batch Size' with the value '1'. The fifth row is 'Learning Rate' with the value '0.00001'. The sixth row is 'Epochs' with the value '3'. The seventh row is 'Loss Function' with the value 'BCE'. A curved arrow points from the 'Train Model' button in the first window to the 'Start Training' button in this window.

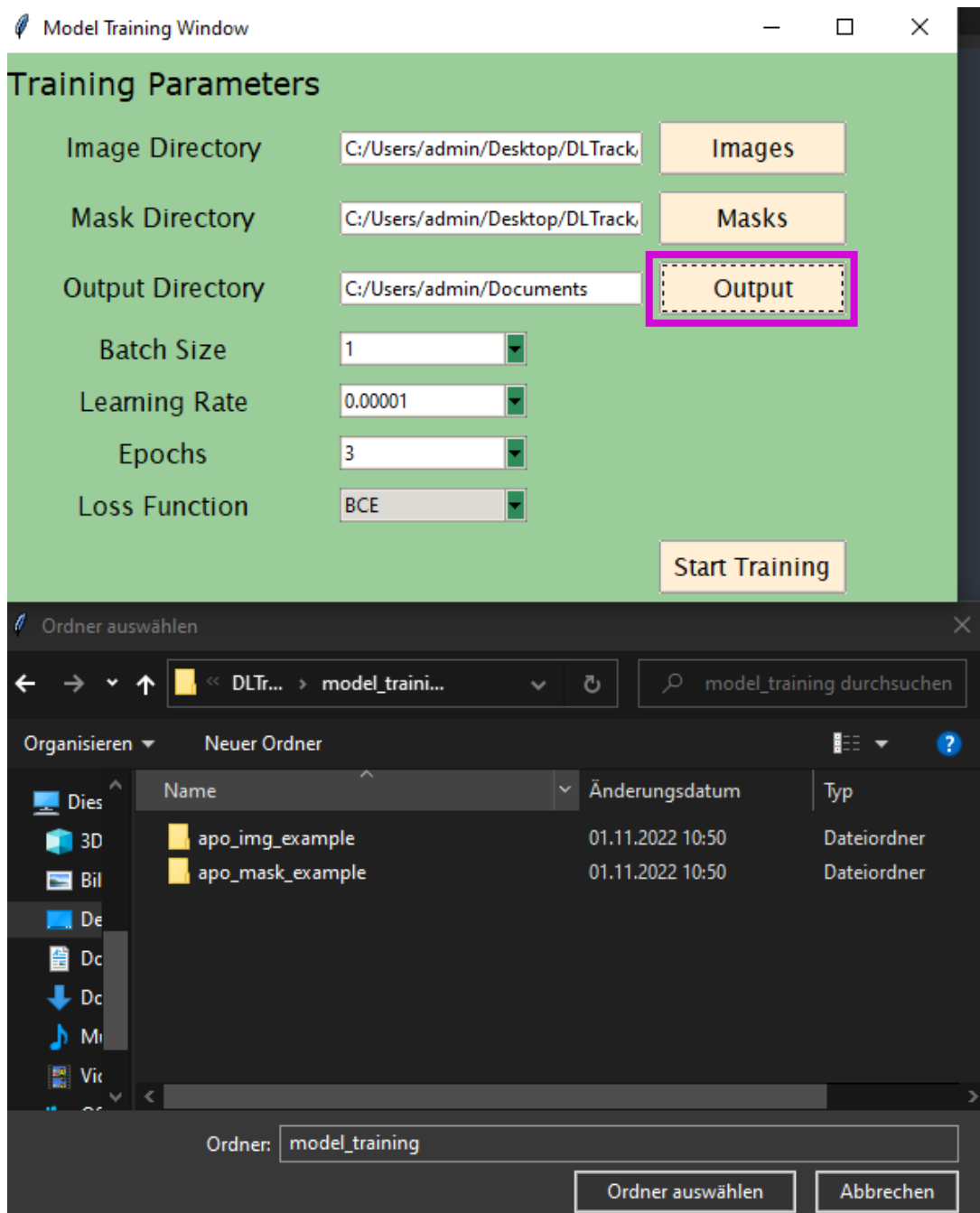
Once the “Model Training window” opened up, you first step is to select the “Image Directory” by clicking the button **Images**. A selection window will appear and you can select the folder containing the **original images**. In this tutorial, select the “DL\_Track\_example/model\_training/apo\_img\_example” folder.



Your next step is to select the “Mask Directory” by clicking the button **Masks**. A selection window will appear and, you guessed correctly, you can select the folder containing the **mask images**. In this tutorial, select the “DL\_Track\_example/model\_training/apo\_mask\_example” folder.



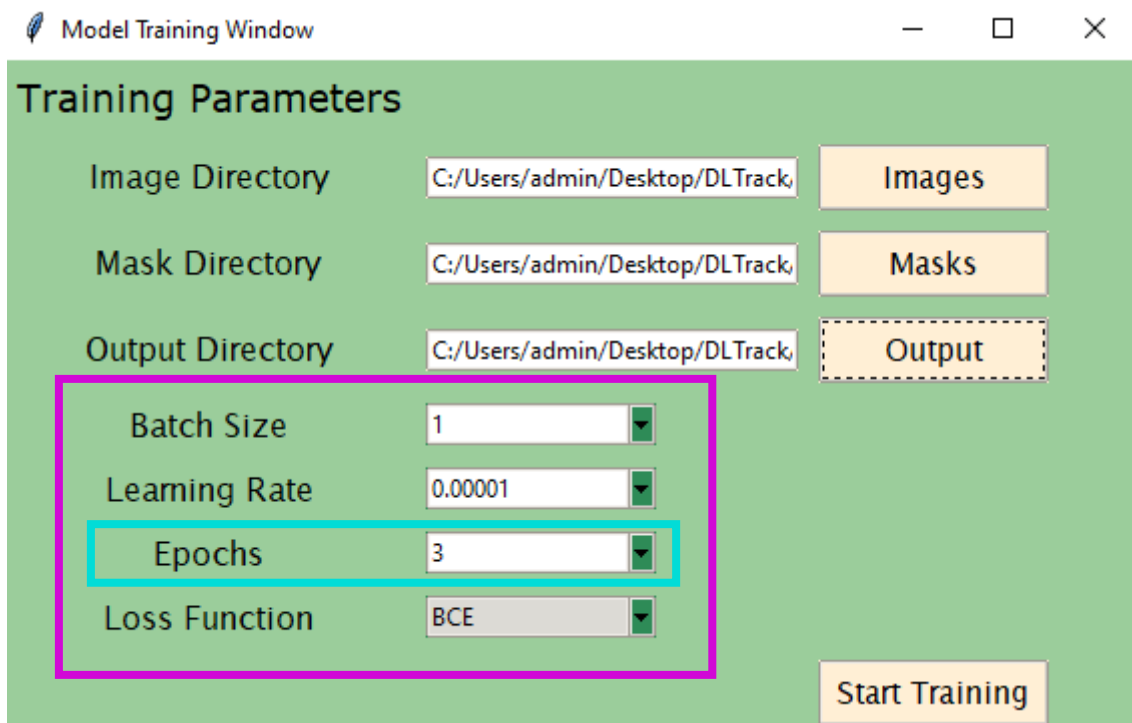
The last directory you need to select for training your own network is the “Output Directory” by clicking the button **Output**. In the **Output** directory, the trained model, the corresponding loss calculation results and a graphic displaying plotting the training epochs against the loss values will be saved. A selection window will appear and you can select any folder you like. In this tutorial, for simplicity reasons, please select the “DL\_Track\_example/model\_training” folder.



Great, you have select all relevant directories for model training! Now we shall take a look at the training parameters on the next page.

### 3. Specifying Training parameters

Now to specifying the **training parameters**. You actually don't have to do anything at this point, just leave the pre-specified selections as they are. They are as suggested by the original papers that introduced the models trained here<sup>1,2</sup> (except for the number of **Epochs**) and proved to result in the best performing models during our tests. If you do not know what these **training parameters** mean, take a look at the videos we mentioned in the introductory text of this chapter. The only thing we have to say is that you must **NEVER** use only three **Epochs** for actual model training. Such a small number of training **Epochs** is only acceptable for demonstration and testing purposes. For actual training of your own neural networks, go with at least 60 **Epochs** (or maybe 42 is the better choice?).

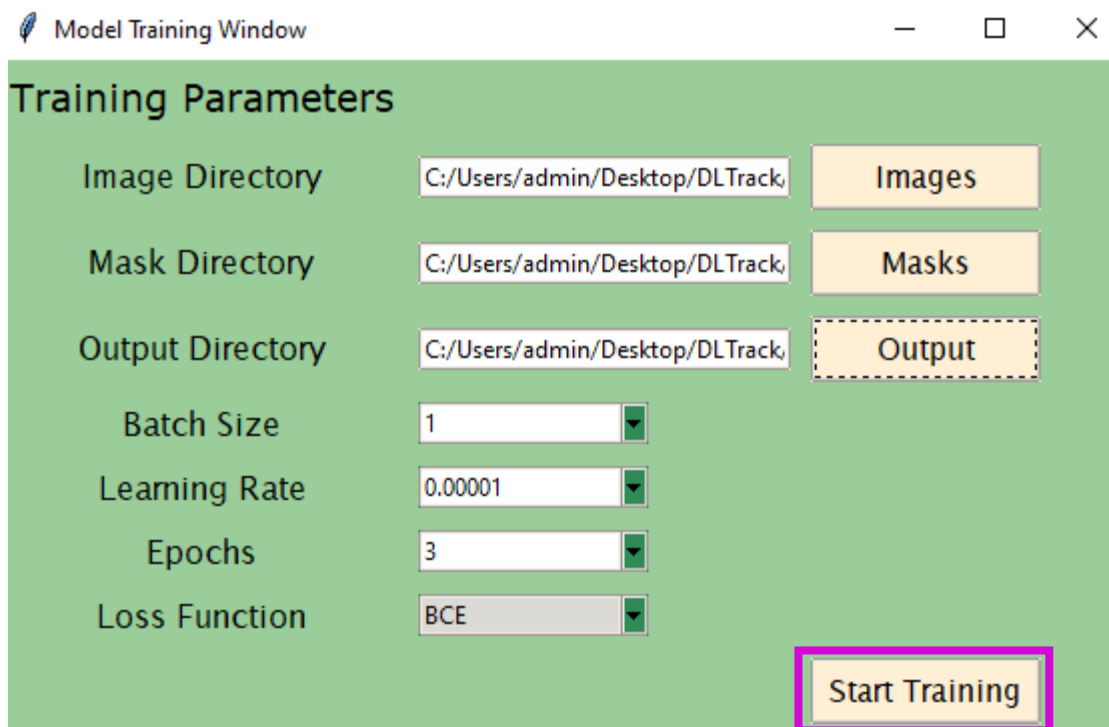


The screenshot shows a window titled "Model Training Window" with a green background. The title bar includes a feather icon, the text "Model Training Window", and standard window controls (minimize, maximize, close). The main area is titled "Training Parameters" and contains several input fields and buttons. On the right side, there are three orange buttons labeled "Images", "Masks", and "Output". The "Output" button is enclosed in a dashed border. A magenta rectangular box highlights a group of five parameters: "Batch Size" (value 1), "Learning Rate" (value 0.00001), "Epochs" (value 3), and "Loss Function" (value BCE). The "Epochs" field is further highlighted with a cyan rectangular box. At the bottom right, there is an orange button labeled "Start Training".

Parameter	Value
Image Directory	C:/Users/admin/Desktop/DLTrack
Mask Directory	C:/Users/admin/Desktop/DLTrack
Output Directory	C:/Users/admin/Desktop/DLTrack
Batch Size	1
Learning Rate	0.00001
Epochs	3
Loss Function	BCE

Take a look at the next page to learn how to continue from here.

The only thing you have left to do for the training process to start is to click the **Start Training** button.



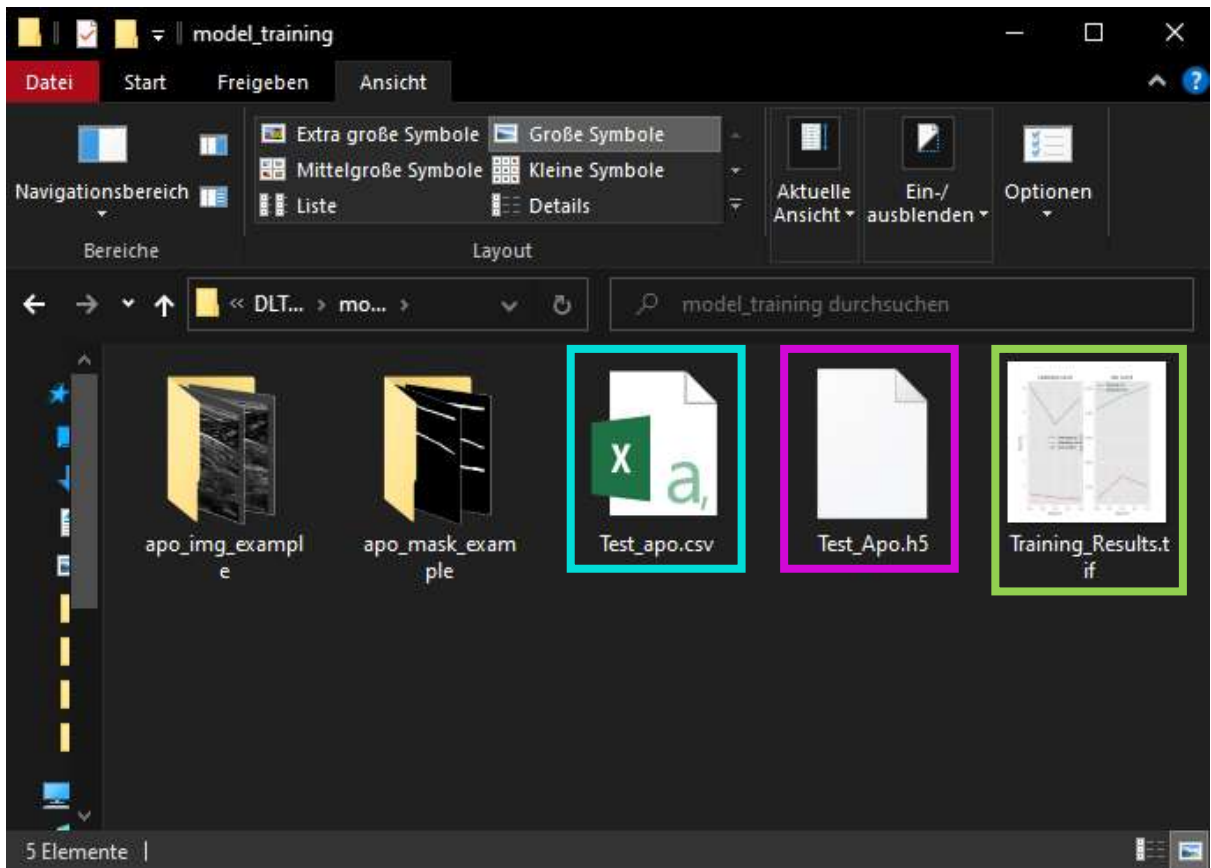
The screenshot shows a window titled "Model Training Window" with a green background. The title bar includes a feather icon, the text "Model Training Window", and standard window controls (minimize, maximize, close). The main area is titled "Training Parameters" and contains the following fields and buttons:

Parameter	Value	Action Button
Image Directory	C:/Users/admin/Desktop/DLTrack	Images
Mask Directory	C:/Users/admin/Desktop/DLTrack	Masks
Output Directory	C:/Users/admin/Desktop/DLTrack	Output
Batch Size	1	
Learning Rate	0.00001	
Epochs	3	
Loss Function	BCE	

At the bottom right, there is a button labeled "Start Training" which is highlighted with a red rectangular border.

During the training process, three messageboxes will pop up. The first one will tell you that the images and masks were successfully loaded for further processing. The second one will tell you that the model was successfully compiled and can now be trained. The last one will tell you that the training process was completed. You do have a choice in each messagebox of clicking "OK" or "Cancel". Clicking "OK" will continue the training process, whereas clicking "Cancel" will be cancelling the ongoing training process.

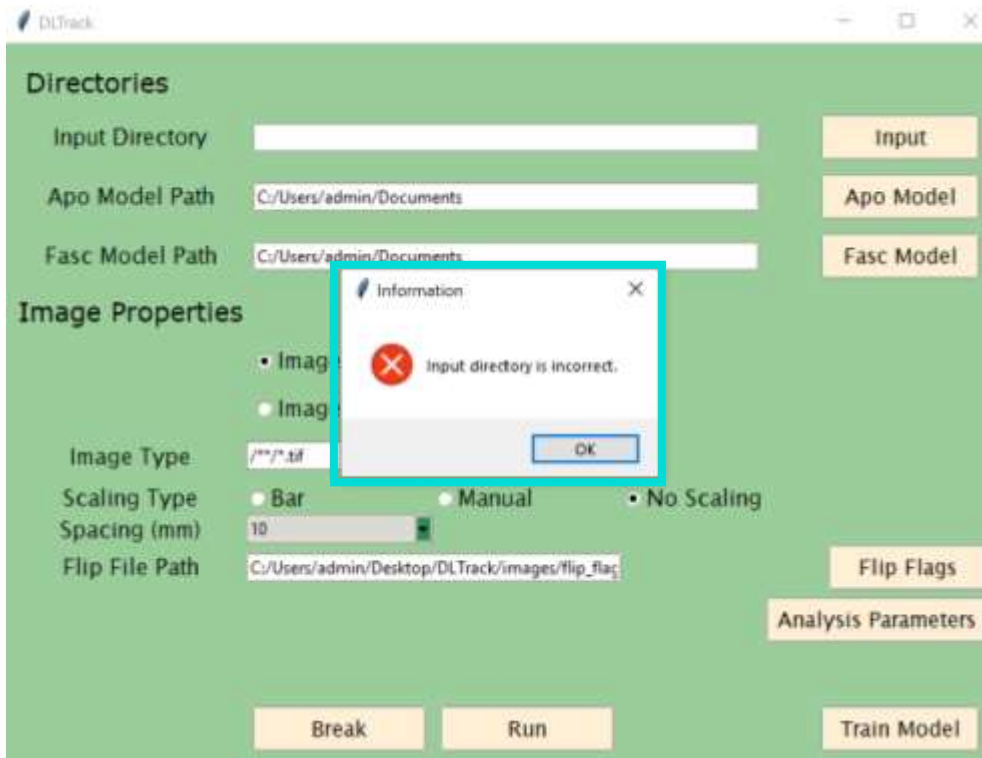
Once the model training process is completed, there will be three new files in your selected output directory. Remember, you specified the “DL\_Track\_example/model\_training” folder as your output directory. These new files are the **trained model** as a Test\_Apo.h5 file, the **corresponding loss values** for each epoch as Test\_apo.csv file and a **graphical representation** of the training process as Training\_Results.tif file.



In case you can open the loss values and the graphical training representation, we congratulate you! You have successfully trained your first own neural network using the DL\_Track package. Awesome! This is also the end of this tutorial, as Training Your Own Networks was the final chapter. Have a look at the closing remarks for a few more things. And, before it is forgotten, there is one more section on the next page. Error handling.

## 4. Error handling

Whenever an error occurs during the analysis process, the DL\_Track GUI will open a **messagebox**. This looks always similar to this:



We tried to formulate these **messageboxes** as concise as possible. Just follow their instructions to fix the error and run the analysis anew. In case an error occurs that is not caught by an error **messagebox**, don't hesitate to report this in the Q&A section in the DL\_Track discussion forum. Please take a look at the "DL\_Track\_bugreport.md" file in this folder how do best do this. Otherwise, you can contact us by email at [paul.ritsche@unibas.ch](mailto:paul.ritsche@unibas.ch), but we would prefer the other way.



# *Closing remarks*

Thanks for checking out the DL\_Track python package tutorial. We hope you were able to enjoy it a bit. Moreover, we hope it was clear, concise and easy to follow. We tried to put our biases aside and to start from scratch. In case we failed to do so at some point and something was not clearly illustrated, please let us know. Don't hesitate to report this in the Q&A section in the [DL\\_Track discussion forum](#). Otherwise, you can contact us by email at [paul.ritsche@unibas.ch](mailto:paul.ritsche@unibas.ch), but we would prefer the other way.