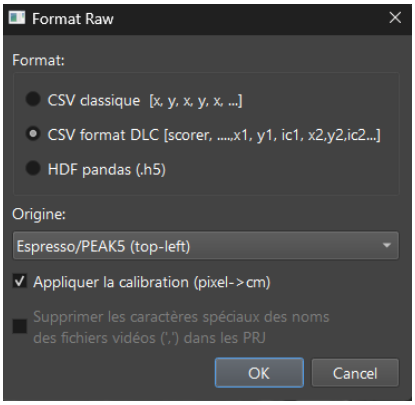
Analyze and Create Labeled video in Deeplabcut

1. Add the video to the deeplabcut project folder: videos
2. Open a conda environnement and type the following commands:
   activate DEEPLABCUT
   ipython
   import deeplabcut
3. To analyze the video, run the command: **deeplabcut.analyze_video(...)**
4. To create a labeled video, run the command:
   deeplabcut.create_labeled_video(...)

Convert the pixel data into cm using Espresso

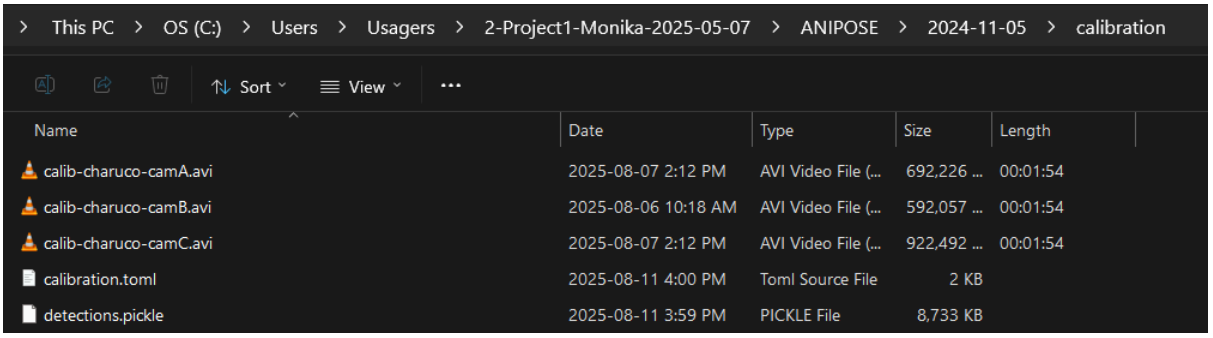
For this step, each video orientation will be done separately.

1. Open Espresso
2. Upload the video in Espresso (File -> Video…)
3. Upload the config file (Model -> Config DLC)
4. Upload the corresponding csv file from the video folder in the deeplabcut project (Import -> CSV coordonnées XYs)
5. Click on the feature “calib” and place the scale on a known 10 cm object (round circle of the precision grip task)
6. Export the converted csv file (Export -> CSV coordonnées XY) while making sure to select the option of conversion from pixel to cm



Calibration of cameras using Anipose

1. Take a calibration video with a charuco board. If needed, a script called ChArUco_board_creation.py can be used to create a charuco board and customize it’s characteristics. If the number of squares or dictionary used to make the motif of the board is changed, the config.toml file must be changed.
2. Add the three calibration videos to the calibration folder of the anipose project and label them like such:



3. Make sure the calibration.toml and detections.pickle file are deleted before running the calibration (they are the desired output of a calibration)
4. Change the date of the folder in which the calibration folder resides (in this example, 2024-11-05)
5. Activate a conda environment where anipose is installed and change the directory (cd) into the one with your project

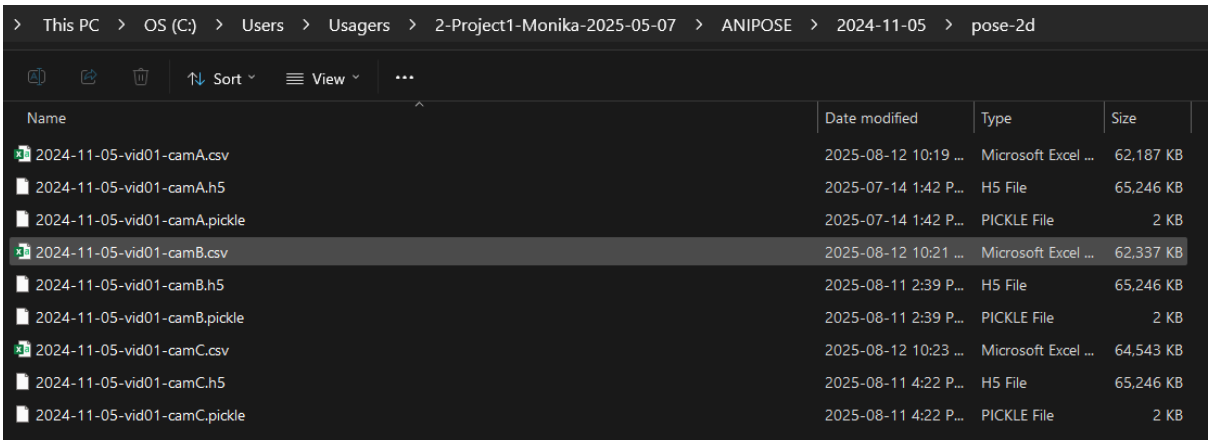
```
(base) C:\Users\Usagers>activate test2

(test2) C:\Users\Usagers>cd 2-Project1-Monika-2025-05-07\ANIPOSE
```

6. Run the command: **anipose calibrate** in the conda prompt

```
(test2) C:\Users\Usagers\2-Project1-Monika-2025-05-07\ANIPOSE>anipose calibrate
Calibrating...
C:\Users\Usagers\2-Project1-Monika-2025-05-07\ANIPOSE\2024-11-05
C:\Users\Usagers\2-Project1-Monika-2025-05-07\ANIPOSE\2024-11-05\calibration\calibration.toml
C:\Users\Usagers\2-Project1-Monika-2025-05-07\ANIPOSE\2024-11-05\calibration\calib-charuco-camA.avi
100%|                                   | 6894/6894 [03:56<00:00, 29.10it/s]
3070 boards detected
C:\Users\Usagers\2-Project1-Monika-2025-05-07\ANIPOSE\2024-11-05\calibration\calib-charuco-camB.avi
100%|                                   | 6894/6894 [02:51<00:00, 40.30it/s]
2522 boards detected
C:\Users\Usagers\2-Project1-Monika-2025-05-07\ANIPOSE\2024-11-05\calibration\calib-charuco-camC.avi
100%|                                   | 6894/6894 [05:57<00:00, 19.29it/s]
5925 boards detected
```

Conversion of XY coordinates into XYZ using Anipose

1. In the folder pose-2d of the ANIPOSE folder, place the .h5 and the .pickle files generated by Deeplabcut for each video, as well as the .csv file for each video generated by Espresso. Change the names of the files so they correspond to the ones in the picture:



C1=camA, C2=camB, C3=camC

2. In the folder videos-raw in ANIPOSE, place the original, unlabeled videos. Change the names to correspond to the picture above.
3. In the conda prompt, run the command: **anipose triangulate**

```
(test2) C:\Users\Usagers\install-Monika\example_ANIPOSE>anipose triangulate
Triangulating points...
C:\Users\Usagers\install-Monika\example_ANIPOSE\2019-08-02\pose-3d\2019-08-02-vid01.csv
100%|                                                  | 87900/87900 [01:20<00:00, 1092.83it/s]
C:\Users\Usagers\anaconda3\envs\test2\lib\site-packages\aniposelib\cameras.py:1253: NumbaWarning:
Compilation is falling back to object mode WITHOUT looplifting enabled because Function "_error_fun_triangulation" failed type inference due to: No implementatio
n of function Function(<function diff at 0x000002018B1DBBE0>) found for signature:

 >>> diff(array(float64, 3d, C), n=int64, axis=Literal[int](0))

There are 2 candidate implementations:
  - Of which 2 did not match due to:
  Overload in function 'np_diff_impl': File: numba\np\old_arraymath.py: Line 3538.
    With argument(s): '(array(float64, 3d, C), n=int64, axis=int64)':
   Rejected as the implementation raised a specific error:
     TypingError: got an unexpected keyword argument 'axis'
  raised from C:\Users\Usagers\anaconda3\envs\test2\lib\site-packages\numba\core\typing\templates.py:791

During: resolving callee type: Function(<function diff at 0x000002018B1DBBE0>)
During: typing of call at C:\Users\Usagers\anaconda3\envs\test2\lib\site-packages\aniposelib\cameras.py (1303)


File "..\..\anaconda3\envs\test2\lib\site-packages\aniposelib\cameras.py", line 1303:
    def _error_fun_triangulation(self, params, p2ds,
        <source elided>
        # temporal constraint
        errors_smooth = np.diff(p3ds, n=n_deriv_smooth, axis=0).ravel() * scale_smooth
        ^
```

```
  @jit(forceobj=True, parallel=True)
   Iteration     Total nfev        Cost      Cost reduction    Step norm     Optimality
       0              1         2.6522e+15                                    9.97e+13
       1              2         6.6763e+14      1.98e+15       1.28e+02       1.25e+13
       2              3         1.6927e+14      4.98e+14       3.62e+04       1.55e+12
       3              4         4.1686e+13      1.28e+14       2.08e+05       1.84e+11
       4              5         8.1537e+12      3.35e+13       5.54e+05       1.93e+10
       5              6         8.5552e+11      7.30e+12       7.20e+05       1.45e+09
       6              7         6.3331e+10      7.92e+11       4.96e+05       8.74e+07
```

4. The output csv file with the XYZ coordinates will be in the newly created folder pose-3d.

Parameters Analysis using VS Code

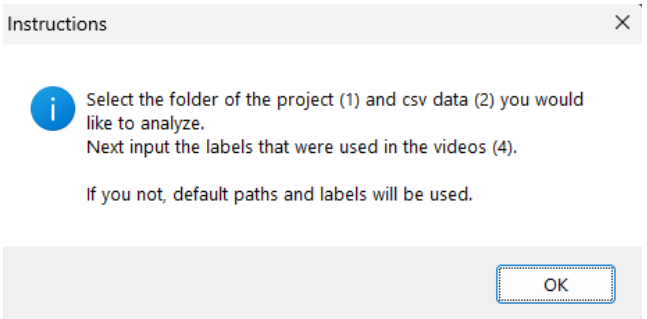
1. Set up a folder with the following files (the exact structure doesn't matter):
   - csv with XYZ,
   - csv with Espresso-converted data from the top camera,
   - a folder with the yaml files for that trial
2. Open Anaconda prompt of the dlcfix environnement

```
(base) C:\Users\Usagers>activate dlcfix

(dlcfix) C:\Users\Usagers>cd C:\Users\Usagers\install-Monika\DLC-Analysis\VS_Code\AnalyseDeeplabcut
```

3. Run the code you want by writing python (chosen_code.py)

```
(dlcfix) C:\Users\Usagers\install-Monika\DLC-Analysis\VS_Code\AnalyseDeeplabcut>python main.py
```

4. A user input window will appear. Press ok and select the right folders.



If running the main.py, use the csv with the xyz coordinates retrieved from anipose. If running the wrist_distance.py, use the csv with the xy coordinates from the top camera. If running the success_rate.py, the csv files used do not matter.
The description of each code's can be found in the Readme in the github.