GTL S43b - Neural Networks Design

The aim of this project is to develop two main Artificial Neural Networks based on a dataset resulting from a huge Design of Experiment (DoE) performed in LCFC Laboratory. This DoE was focused on the analysis of operators' behavior in the same production system in two situations: in the real world and in the Virtual Reality (VR) one. Several candidates (18 people) performed 80 parts by batch of 10, half with the real system and the second half in the virtual one. For both situations, the postures of operators were measured and collected thanks to a motion capture suit composed by 18 sensors and able to snap 60 frames per second.

Figure 1: Motion capture acquisition in the two working situations

The technology used (inertial accelerators) is not free from limits and drawbacks: the sensors can diverge during the measurement, more if the operator is not moving (the disturbances and noise becoming more important than joints displacement).

This work could be carried out by team of **max** 2 Students. Your program and report must be uploaded in SAVOIR in the dedicated section before **25th October (18:00)**. To complete this project, you can use Python programming language with several libs (*matplotlib*, *numpy, tensorflow, keras, scikit-learn…*).

Data set description

The provided dataset (as csv file in SAVOIR) is the result of this DoE. This file is split into three main sections:

- The **description of the worker and the working situation** (operator characteristics: height, sex and socio-professional category, type of condition: real or VR, ID of part manufactured, from 1 to maximum 80).
- The **quality of the posture** measurement, coded as an integer:
  - 1: the posture is considered free from sensor issue,
  - 0: the posture cannot be considered since a deviation was identify.
- The **posture characteristics**. To do so, for each main task of the assembly process composed by 5 (as illustrated in Figure 2) several statistics (min, max, average, median and standard deviation) are given for the main joints of the upper body during these working steps.

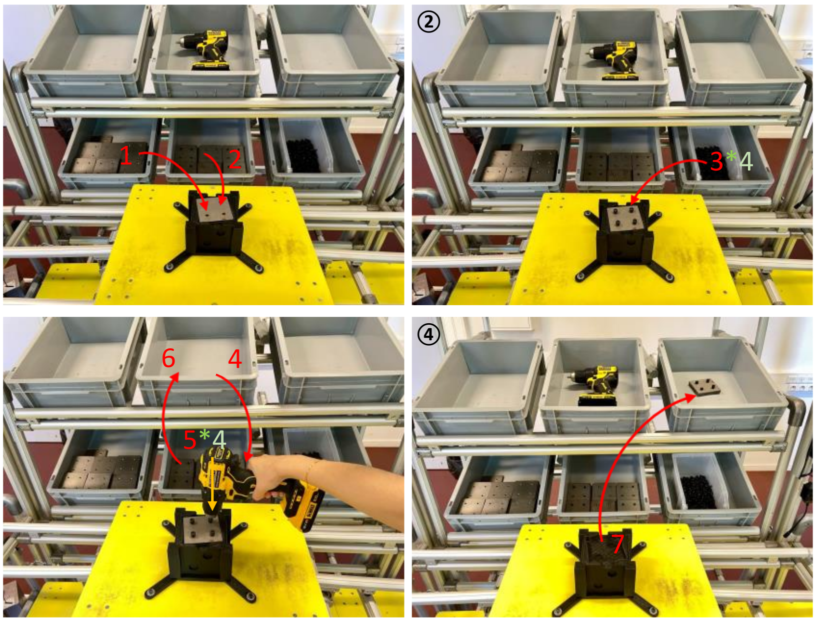

Figure 2: The for main tasks (picking 2 plates, picking 4 screws, screwing and unpicking the final product)

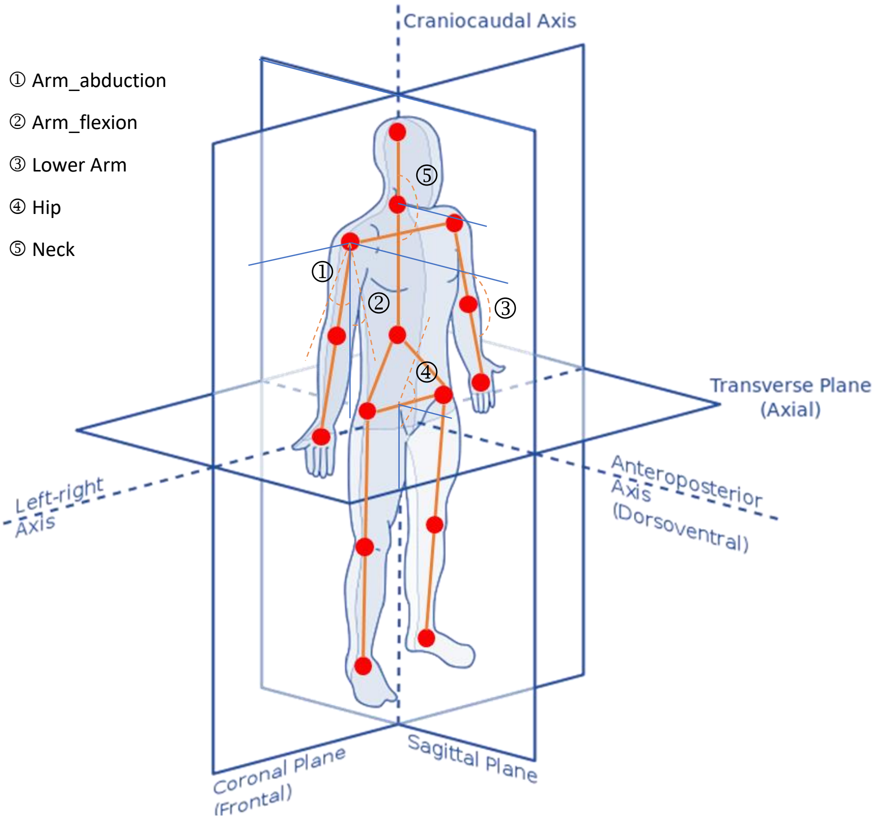

Figure 3 : Identification of the angles considered in the dataset

Of course, the goal is not to consider the whole dataset at once and several smaller networks are to be designed for specific objectives. The next section details these objectives and the work you have to performed.

Work to do

Three main models are expected:

- A first one that **identifies**, regarding the minimum of data, which **postures cannot be considered relevant** regarding human biomechanics.
- A second one that **identifies**, with minimum data, if the part was assembled on the **real** production system **or** on the **virtual** one.
- A last one that **predicts** what will be **joint angles** (you can select the aggregation metrics you considered the best one) regarding the inputs: operator height, sex, professional category, the working situation and the number of parts already performed (fatigue).

The two first models are then classification models and the last one is a regression one. For all of them select your input data and not consider all 200 parameters (since these ones are partially correlated being a statistical metric of the same measures): the training process will take too long and the size of your model is too big compared to the dataset size (risk of overfitting).

If need be, you can split the analysis not by considering all working tasks together, but one by one (the postures differ strongly between them).

In addition to your code and program, please explain and justify your conclusion in a report underlining what's the best configuration you found for each of the three objectives proposed.
Last tip, if you want to identify which type of statistical parameter (min, max, mean, median, standard deviation) is the most relevant, don't hesitate to compare them with a draft model. For this one, you can make a sensitivity analysis too. In addition, if you want you can design a program that tests several configurations of network to identify the most relevant one.