**HAPP-Assignment 0: Human Action Recognition using Machine Learning**

1. **Language & Compilation**

The program is written in **Python** and compiled into .exe files using **PyInstaller**. To compile each .py file into a standalone .exe file:

pyinstaller --onefile T1\_plot.py

pyinstaller --onefile T2\_training\_data\_RMS.py

pyinstaller --onefile T3\_plot\_3D.py

pyinstaller --onefile T4\_test\_data\_RMS.py

pyinstaller --onefile T5\_KNN.py

Make sure the following packages are installed:

pip install numpy matplotlib scikit-learn

1. **How to use**

* Ensure the following folders are present in the current directory:
  + act01, act02, act03: for training data
  + test: for testing data
* Run the corresponding .exe file to perform each task (recommended method).
* Alternatively, you may run .py files manually from the source\_code/ folder.
* **T1.py Tip**: If you do not want the graphs to pop up during execution, you can **edit line 76** in source\_code/T1\_plot.py:

process\_training\_data(training\_folders, show\_plot=False)

1. **Result Details**

* T1\_plot.exe
  + Displays one graph at a time (21 total), each must be closed to continue.
  + Saves all graphs into a folder named T1\_plots in the current directory.
* T2\_training\_data\_RMS.exe
  + Shows calculated RMS values in the terminal.
  + Outputs T2\_training\_features.csv in the current directory.
* T3\_plot\_3D.exe
  + Displays a 3D scatter plot of the RMS features.
  + Saves the image in the current directory.
* T4\_test\_data\_RMS.exe
  + Like T2, outputs T4\_test\_features.csv in the current directory.
* T5\_KNN.exe
  + Performs KNN classification and outputs the result in T5\_prediction\_result.txt.