Project : Gesture classification technique based on multiple EMG datasets

Name

December 23, 2025

1 Database Analysis

- **Data Loading library:**

```
import scipy.io as sio
import matplotlib.pyplot as plt
import numpy as np
import os
import pandas as pd
import csv
import sys
import h5py
```

1.1 Ninapro Dataset

- **Name:** Ninapro (DB5)
- **Data Type:** 16 EMG channels, sampling rate 200Hz, 18 exercises
- **Data shape of one file:** 179,901 rows × 16 cols
- **Exercises in one file:** 17
- **Data Loading Code:**

```
ninapro_df = pd.DataFrame()
for i in range (1,11):
    adress = f"ninapro_db5/ninapro_db_{i}/S{i}_E2_A1"
    filename = adress
    mat = sio.loadmat(filename)
    emg = mat['emg']
    Restimulus = mat['restimulus']
    rerepetition = mat['rerepetition']
    df_emg = pd.DataFrame(emg)
    df_Restimulus = pd.DataFrame(Restimulus)
    df_rerepetition = pd.DataFrame(rerepetition)
    df = pd.concat([df_emg, df_Restimulus], axis=1)
    df = pd.concat([df, df_rerepetition], axis=1)
    df.columns = ['emg1', 'emg2', 'emg3', 'emg4', 'emg5', '
        emg6', 'emg7', 'emg8', 'emg9', 'emg10', 'emg11', '
        emg12', 'emg13', 'emg14', 'emg15', 'emg16', '
        Restimulus', 'rerepetition']
    ninapro_df = pd.concat([ninapro_df, df])
```

1.2 MoveR/Nature Dataset

- **Name:** Nature
- **Data Type:** 16 EMG channels, sampling rate 2000Hz, 6 exercises
- **Exercise Mapping:** 1=6, 2=18, 3=7, 4=5, 5=19, 6=0
- **Data shape of one file:** 16 rows × 8,000 cols
- **Exercises in one file:** 1

```
grasp_mapping = {
    1: 6,
    2: 18,
    3: 7,
    4: 5,
    5: 19,
    6: 0
}

nature_df = pd.DataFrame()
for i in range(1, 9):
    for j in range(1,3):
        for k in range(1,3):
            filename = fr’nature_data\data\participant_{i}\
                participant{i}_day{j}_block{k}\emg_data.hdf5’
            data_parame = pd.read_csv(fr’nature_data\data\
                participant_{i}\participant{i}_day{j}_block{k
                }\trials.csv’)
            nature_data = h5py.File(filename, ’r’)
            data_parame[’grasp’] = data_parame[’grasp’].
                replace(grasp_mapping)
            for l in range(0, 150):
                df = pd.DataFrame(np.array(nature_data[f"{l}
                    "]))
                df=df.transpose()
                df[’Restimulus’] = ’’
                df[’Restimulus’] = data_parame[’grasp’].iloc
                    [l]
                nature_df = pd.concat([nature_df, df], axis
                    =0)
                print(f"nature_df{i}{j}{k}{l}␣finished")
nature_df.columns = [’emg1’, ’emg2’, ’emg3’, ’emg4’, ’emg5’,
     ’emg6’, ’emg7’, ’emg8’, ’emg9’, ’emg10’, ’emg11’, ’emg12
    ’, ’emg13’, ’emg14’, ’emg15’, ’emg16’, ’Restimulus’]
# nature_df.to_csv(’nature_df.csv’, index=False)
```

1.3 Kaggle Dataset

- **Name:** Kaggle (EMG Gesture Dataset)
- **Data Type:** 8 EMG channels, sampling rate 985Hz, 12 exercises
- **Data shape of one file:** 8 rows × 9,980 cols
- **Exercises in one file:** 1

Abbr.	Full Name	ID	Abbr.	Full Name	ID
TU	Thumb Up	1	HO	Hand Open	5
IDX	Index	7	WE	Wrist Extension	14
RA	Right Angle	20	WF	Wrist Flexion	13
PCE	Peace	2	UD	Ulnar Deviation	16
IL	Index Little	21	RD	Radial Deviation	15
HC	Hand Close	6			

Table 1: Kaggle Dataset Gesture Mapping

2 Channel-wise Encoder + Attention Pooling (CWE-AP)

2.1 Mathematical Formulation