SIGNAL PROCESSING
FOR DATASCIENCE

# EPILEPSY PREDICTION

TUH EEG SEIZURE CORPUS DATASET


Presented by:

Messaadi Ouday & Cherni Oussema

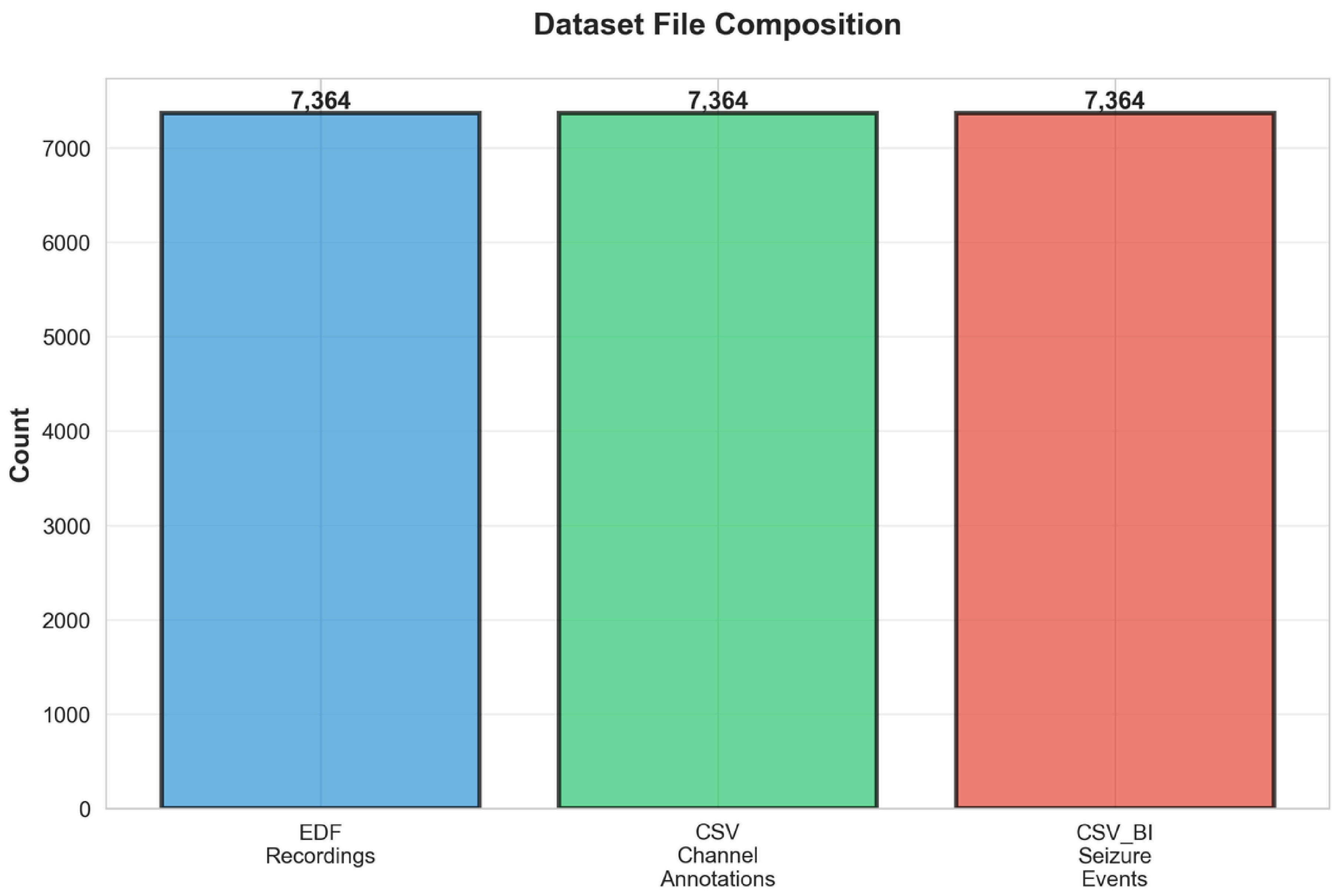

# PLAN

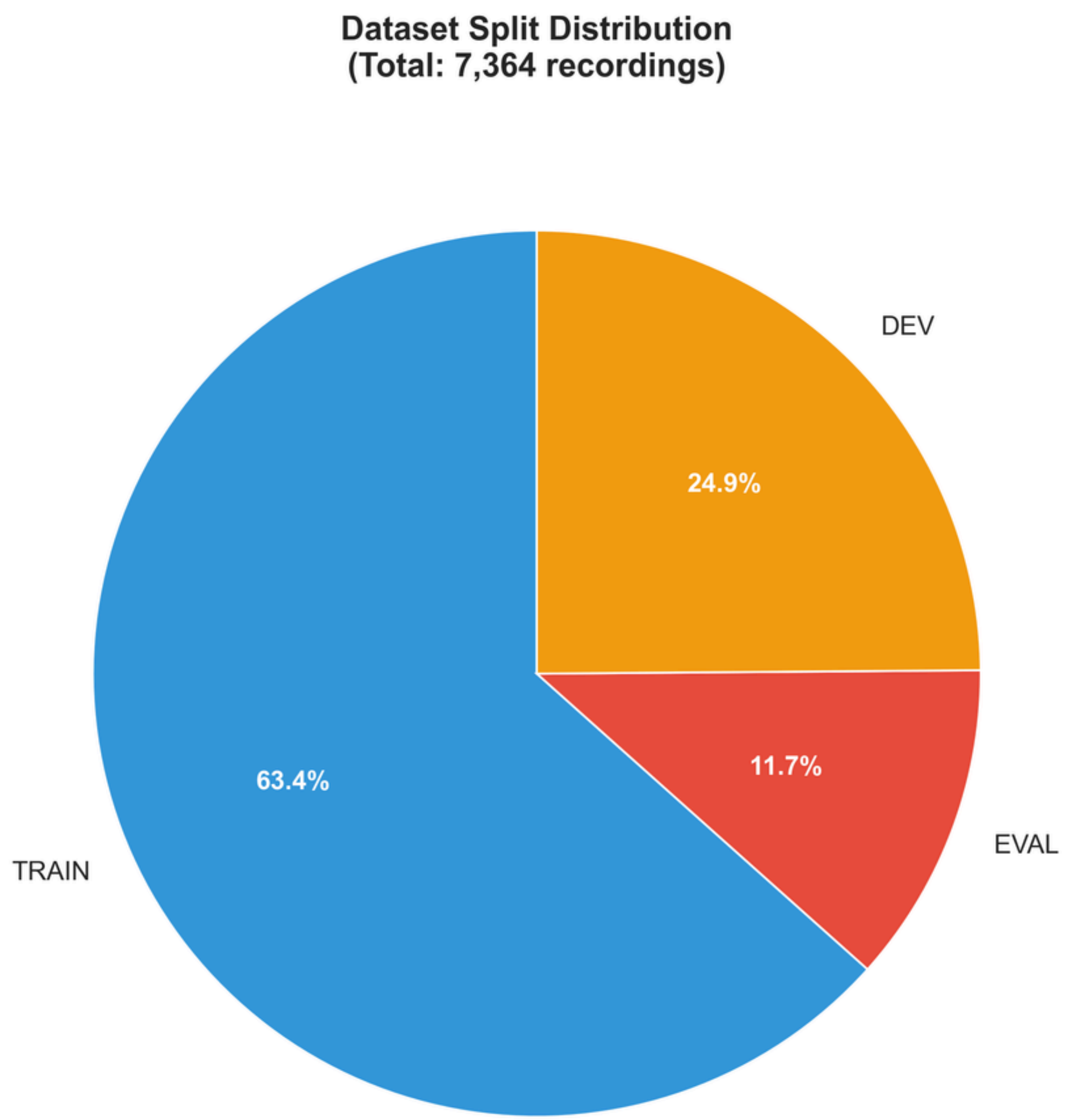
1. Statistical Analysis
2. Data Visualization
3. Data Preprocessing
4. CNN Model
5. Results

# REFERENCES

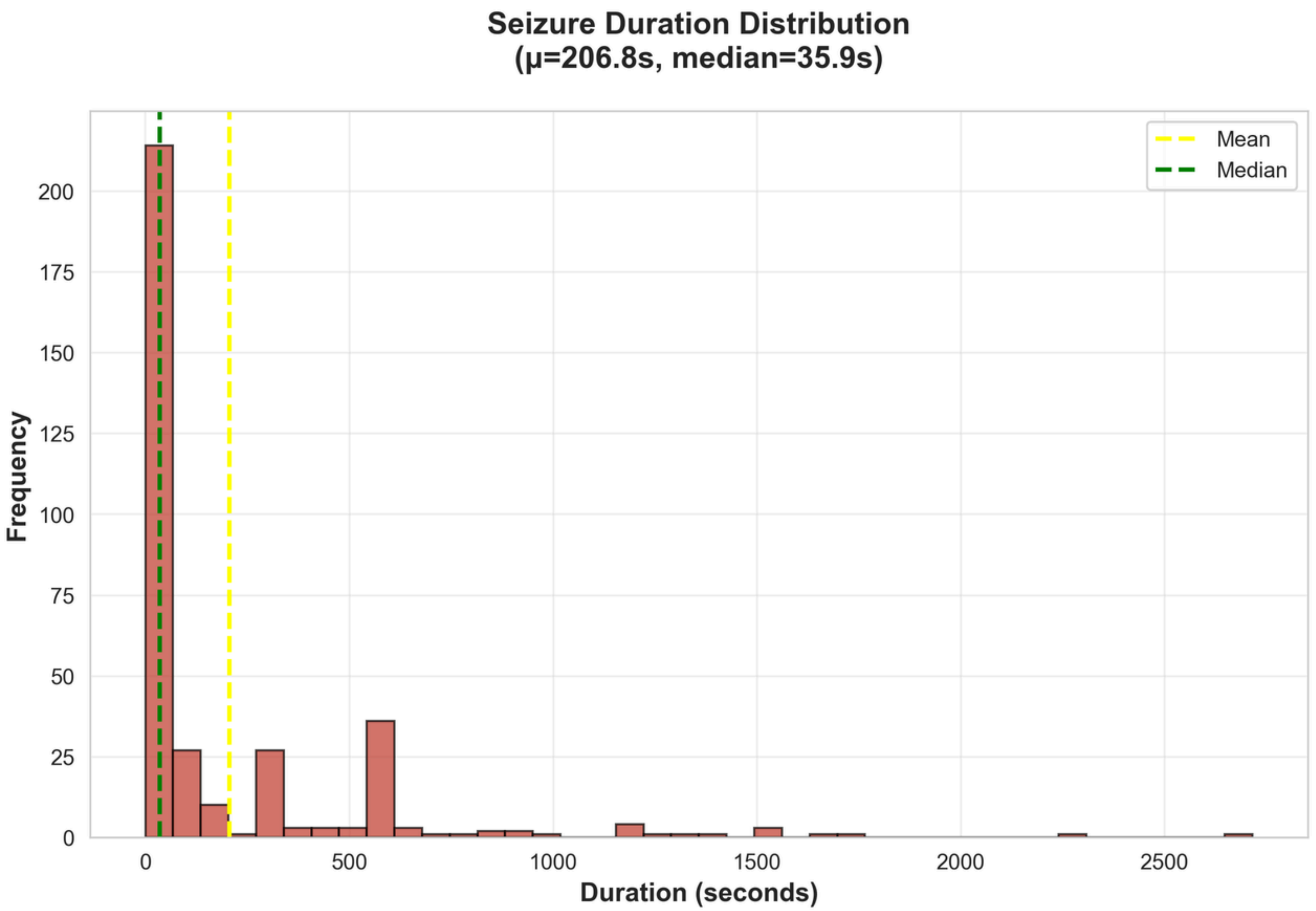
| Reference | Method | Relevance to Our Work |
|---|---|---|
| **Deep learning based automatic seizure prediction with EEG time-frequency representation**<br>X. Dong et al., 2024<br>ScienceDirect | • Multi-channel EEG from clinical databases<br>• Stockwell Transform → 2D time-frequency images<br>• Multi-Channel Vision Transformer (MViT)<br>• Pre-ictal vs inter-ictal prediction (~97% accuracy) | Very similar pipeline: EEG → time-frequency images (like your spectrograms) → CNN/vision model. Modern example of time-frequency + deep learning for seizure prediction. |
| **EEG-based seizure prediction via Transformer guided CNN (TGCNN)**<br>C. Li et al., 2022<br>ScienceDirect | • STFT for EEG spectrograms<br>• Hybrid architecture: CNN (local features) + Transformer (long-term dependencies)<br>• Tested on CHB-MIT & Kaggle (~91.5% sensitivity, low false alarms) | Same approach: EEG spectrograms + CNN. Adds Transformer layer for global context. Excellent reference if you want to improve your CNN with attention mechanisms. |
| **Semi-dilated convolutional neural networks for epileptic seizure prediction**<br>R. Hussein et al., 2021<br>ScienceDirect | • EEG → scalograms (wavelet-based time-frequency)<br>• 2D CNN with semi-dilated convolutions<br>• Adapted for elongated images (more temporal than frequency resolution)<br>• Pre-ictal vs normal prediction | EEG → time-frequency image → CNN. Shows how to adapt CNN architecture to spectrogram shape (filter sizes, dilation). Good inspiration for CNN design choices. |
| **Convolutional neural networks for seizure prediction using intracranial and scalp EEG**<br>T. Truong et al., 2018<br>ScienceDirect | • Intracranial + scalp EEG (CHB-MIT + other datasets)<br>• CNN for pre-ictal vs inter-ictal prediction<br>• Sliding temporal windows<br>• Patient-specific vs generic models comparison | Same goal: seizure prediction with CNN. Discusses inter-patient generalization, which is crucial when working with large datasets like TUH TUSZ. |

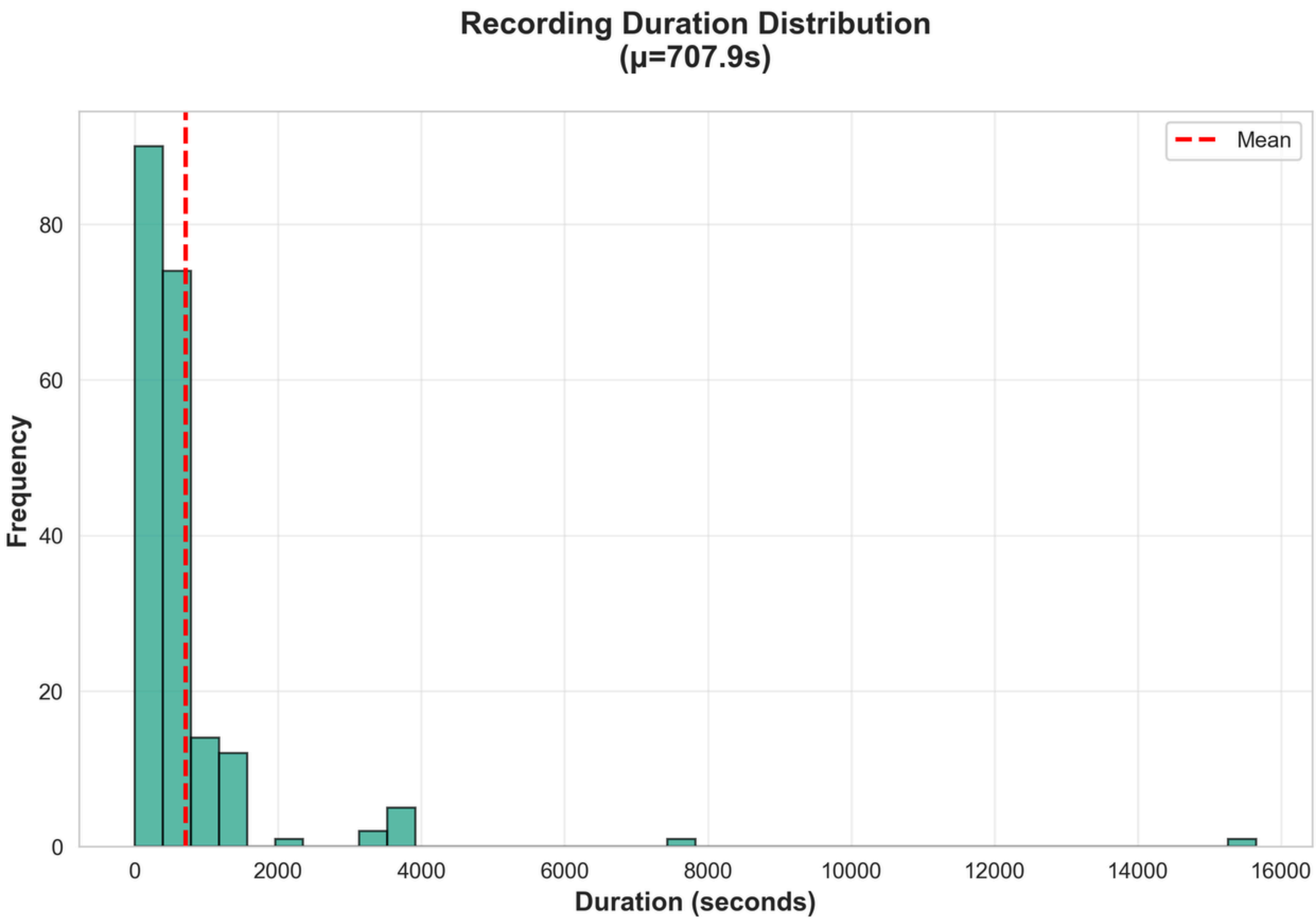
# STATISTICAL ANALYSIS OVER THE TUH DATASET



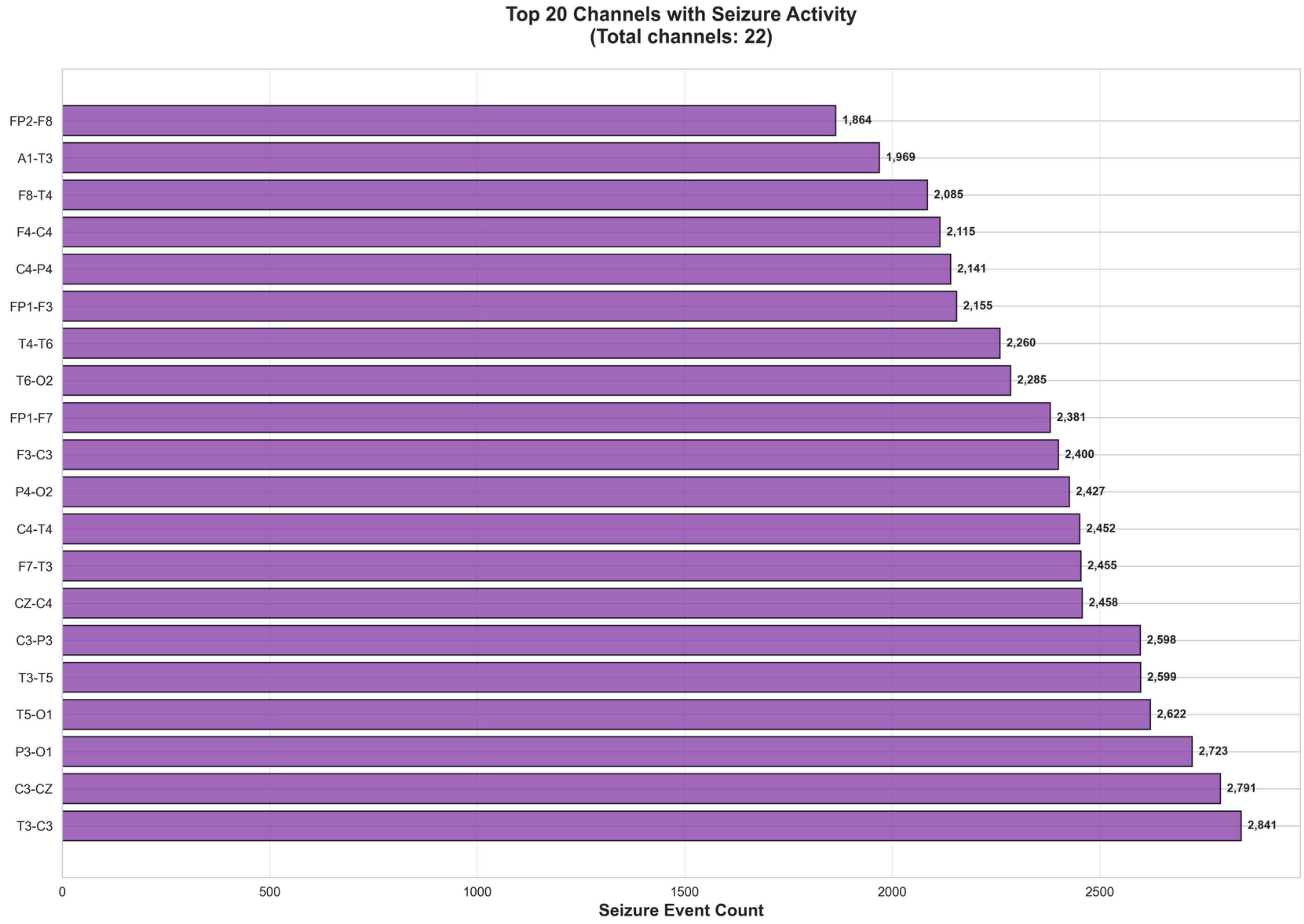



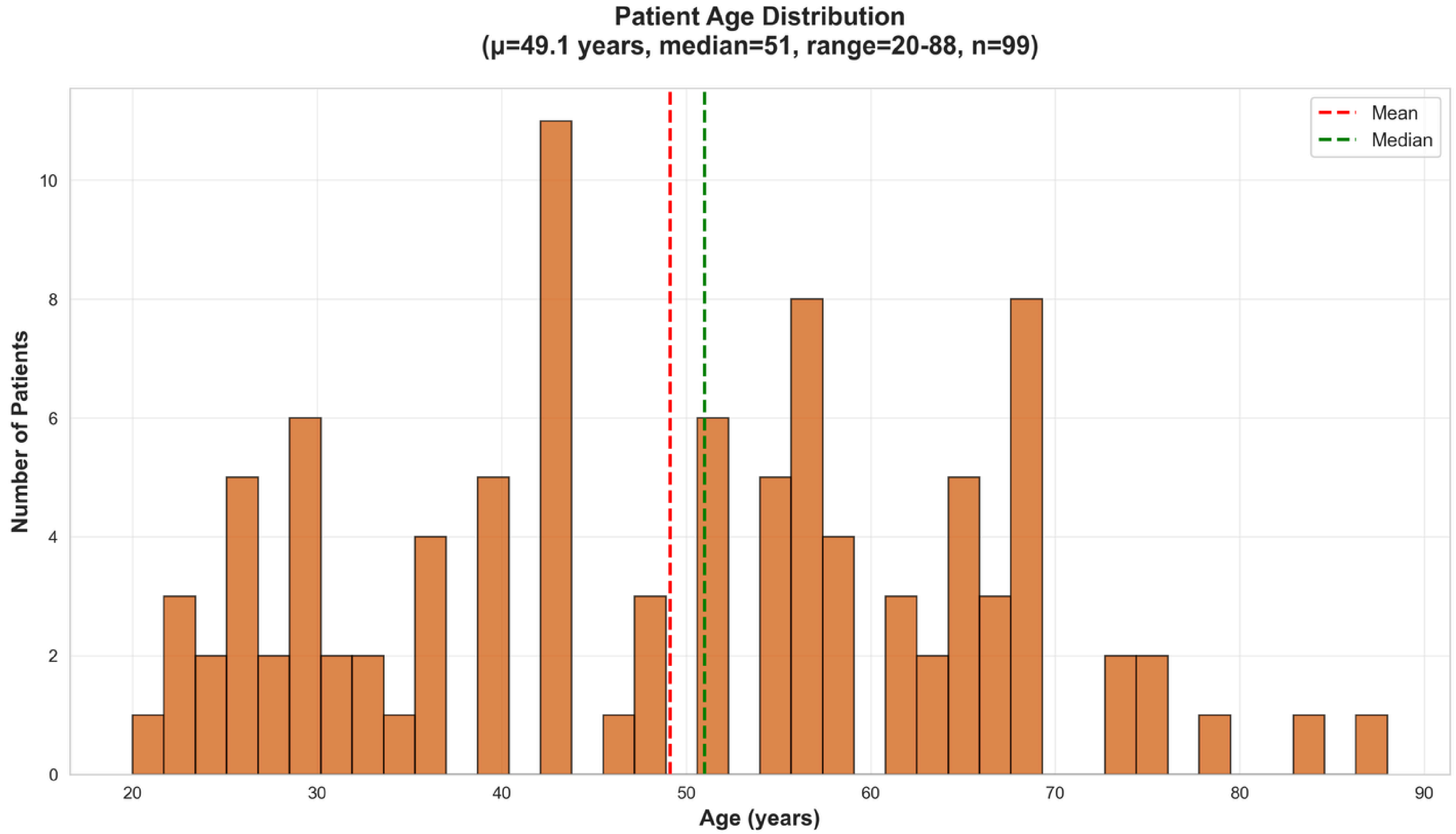
# STATISTICAL ANALYSIS OVER THE TUH DATASET



# STATISTICAL ANALYSIS OVER THE TUH DATASET



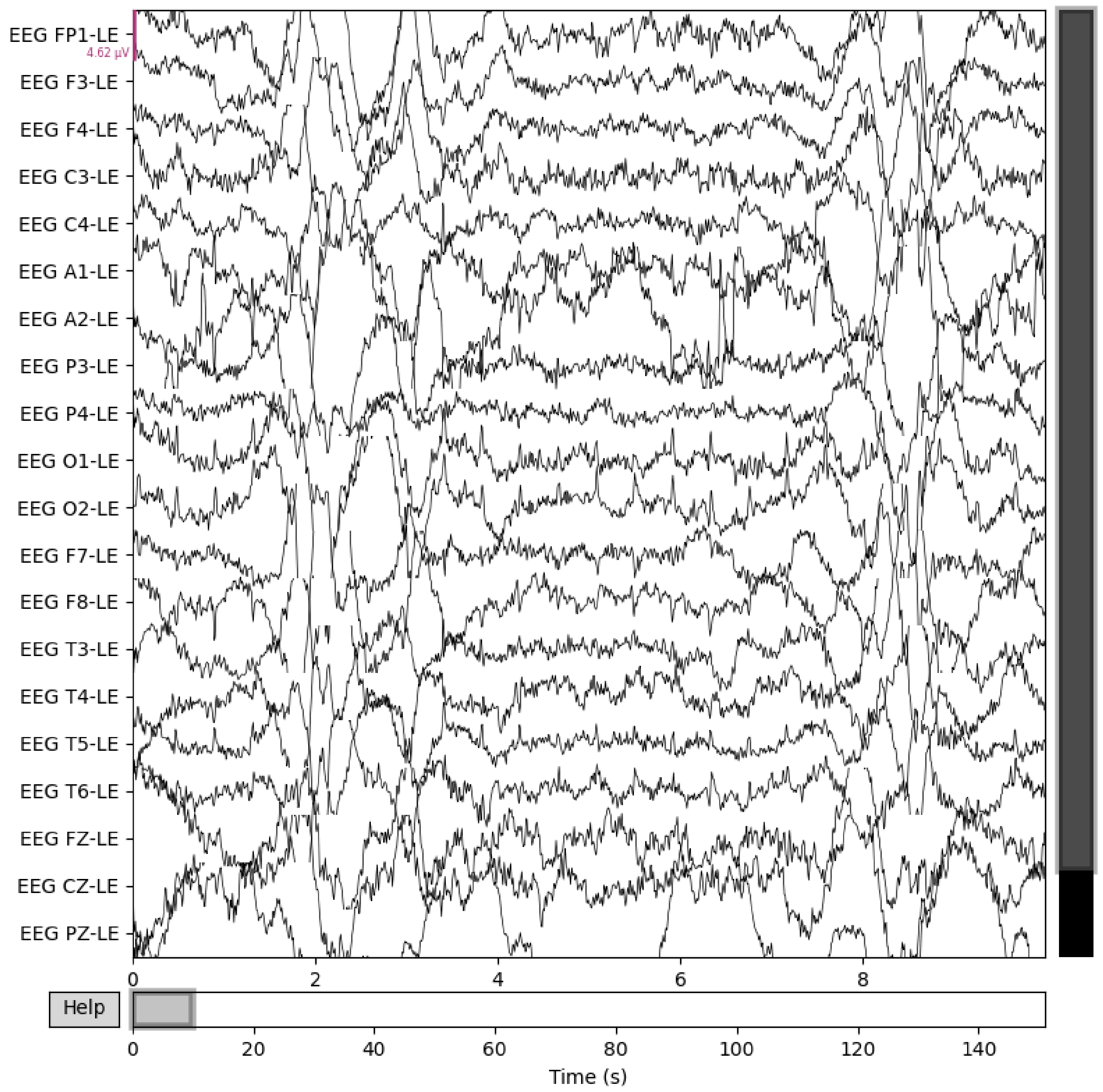
# STATISTICAL ANALYSIS OVER THE TUH DATASET

| DATASET OVERVIEW | |
|---|---|
| Total Files Scanned | 22,093 |
| Valid EDF Files | 7,364 |
| Valid CSV Files | 7,365 |
| Valid CSV_BI Files | 7,364 |
| Invalid Files | 0 |
| | |
| PATIENT & SESSION INFO | |
| Unique Patients | 675 |
| Unique Sessions | 221 |
| Avg Recordings/Patient | 10.91 |
| Avg Sessions/Patient | 2.43 |
| | |
| DATA SPLITS | |
| Train Recordings | 4,667 |
| Eval Recordings | 865 |
| Dev Recordings | 1,832 |
| | |
| SEIZURE INFORMATION | |
| Total Seizure Events | 9,943 |
| Unique Seizure Types | 8 |
| Avg Seizure Duration | 241.80s |
| Median Seizure Duration | 59.98s |
| | |
| RECORDING DETAILS | |
| Avg Recording Duration | 720.4s |
| Most Common Sampling Rate | 256 Hz |
| Avg Channels/Recording | 30.5 |
| Unique Channel Names | 99 |

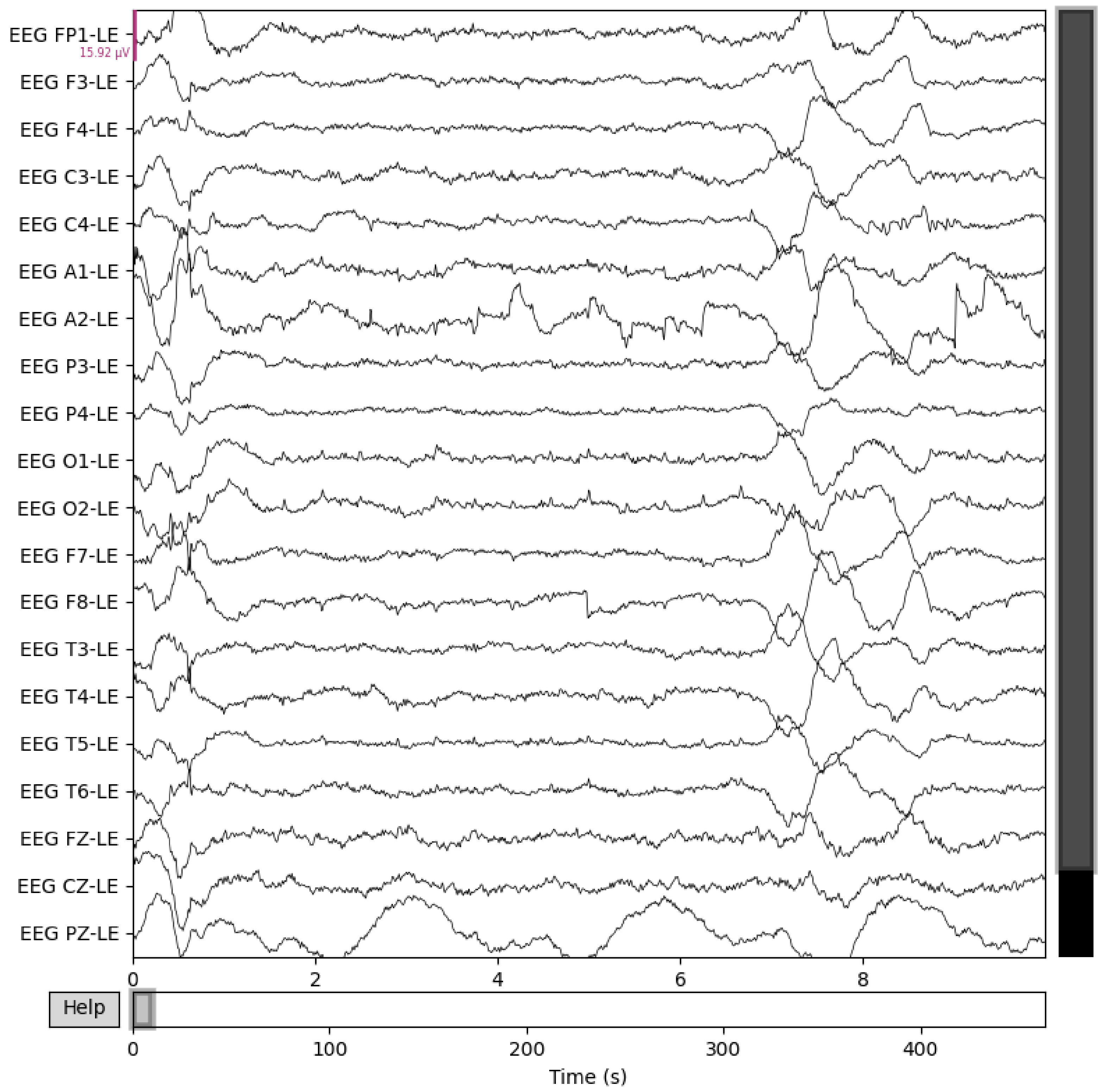
# DATA VISUALISATION USING MNE OVER THE SAME PATIENT DURING SEIZURE VS NON-SEIZURE RECORDING

# EEG SIGNAL EXAMPLES: NON-SEIZURE VS SEIZURE

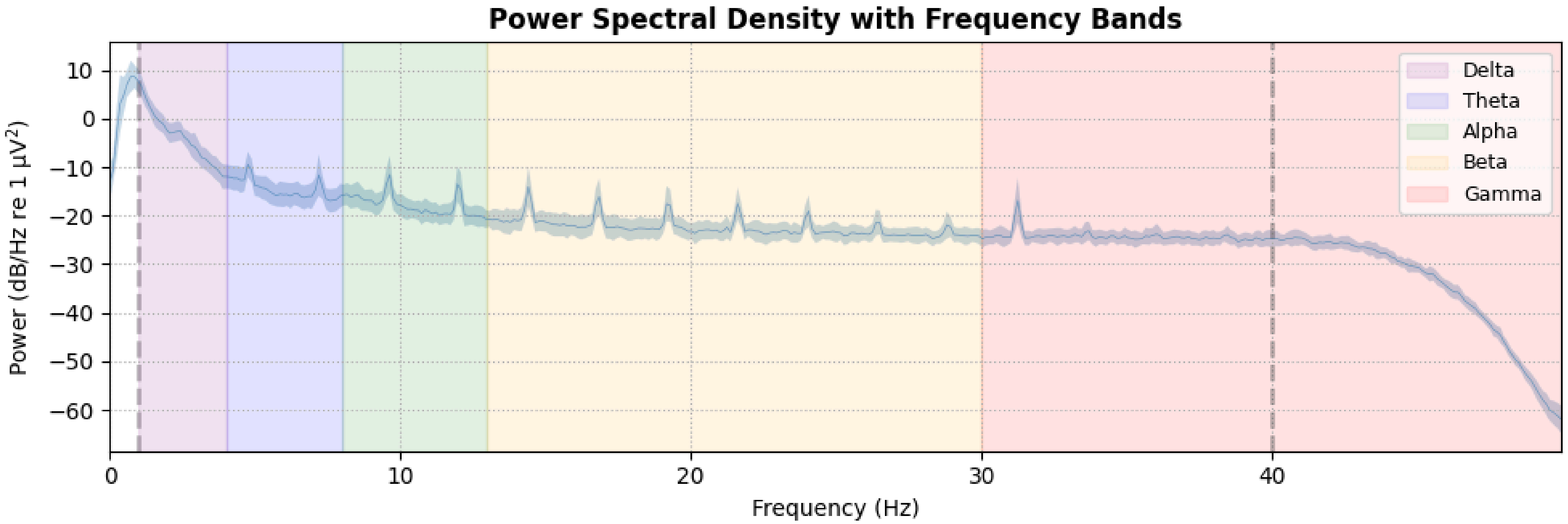
## non-seizure



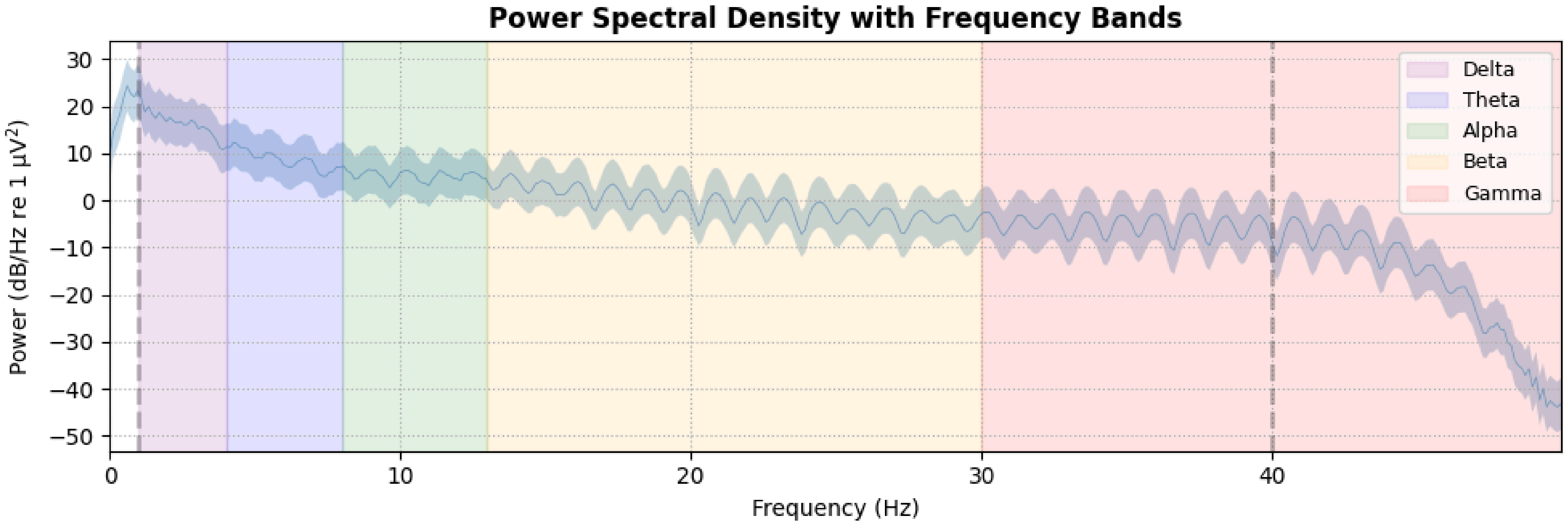

## seizure



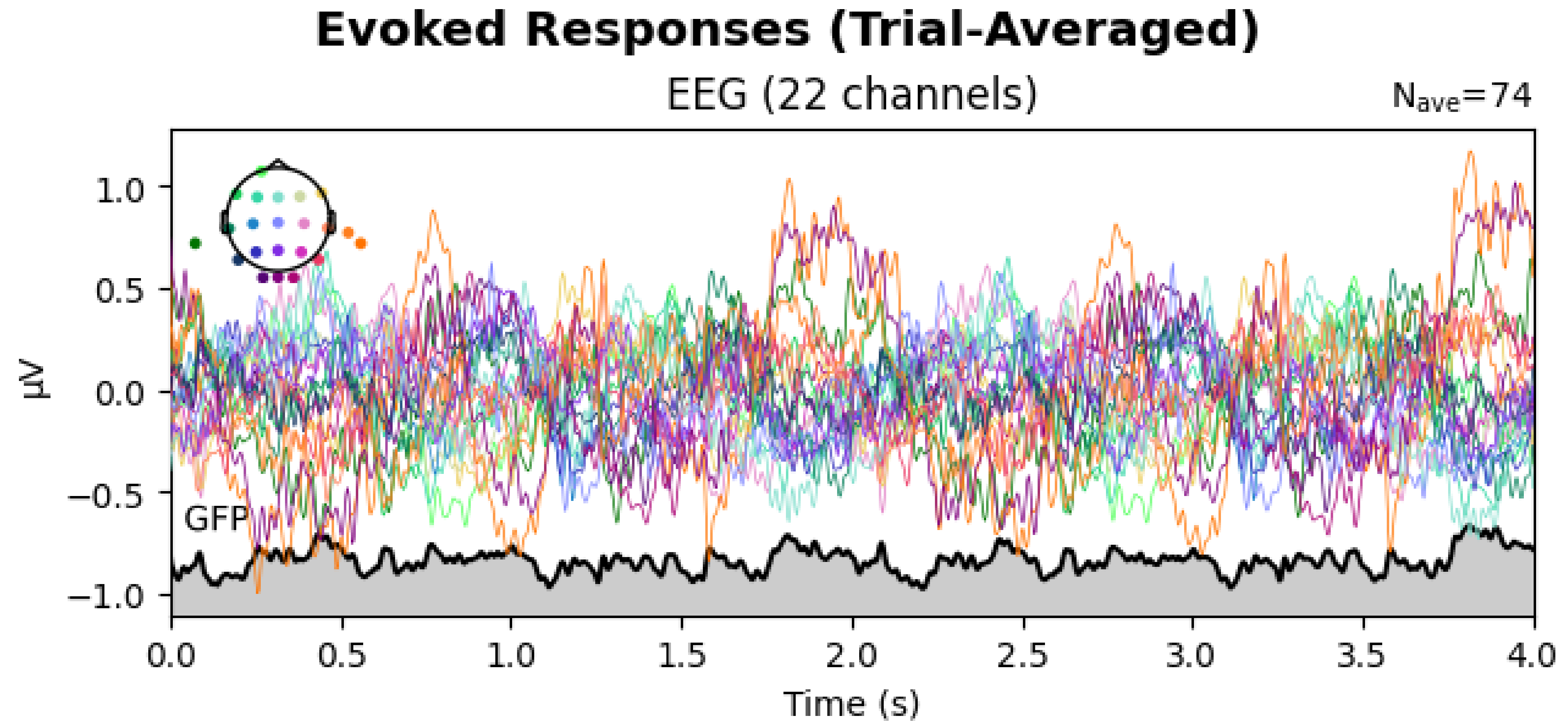
non-seizure



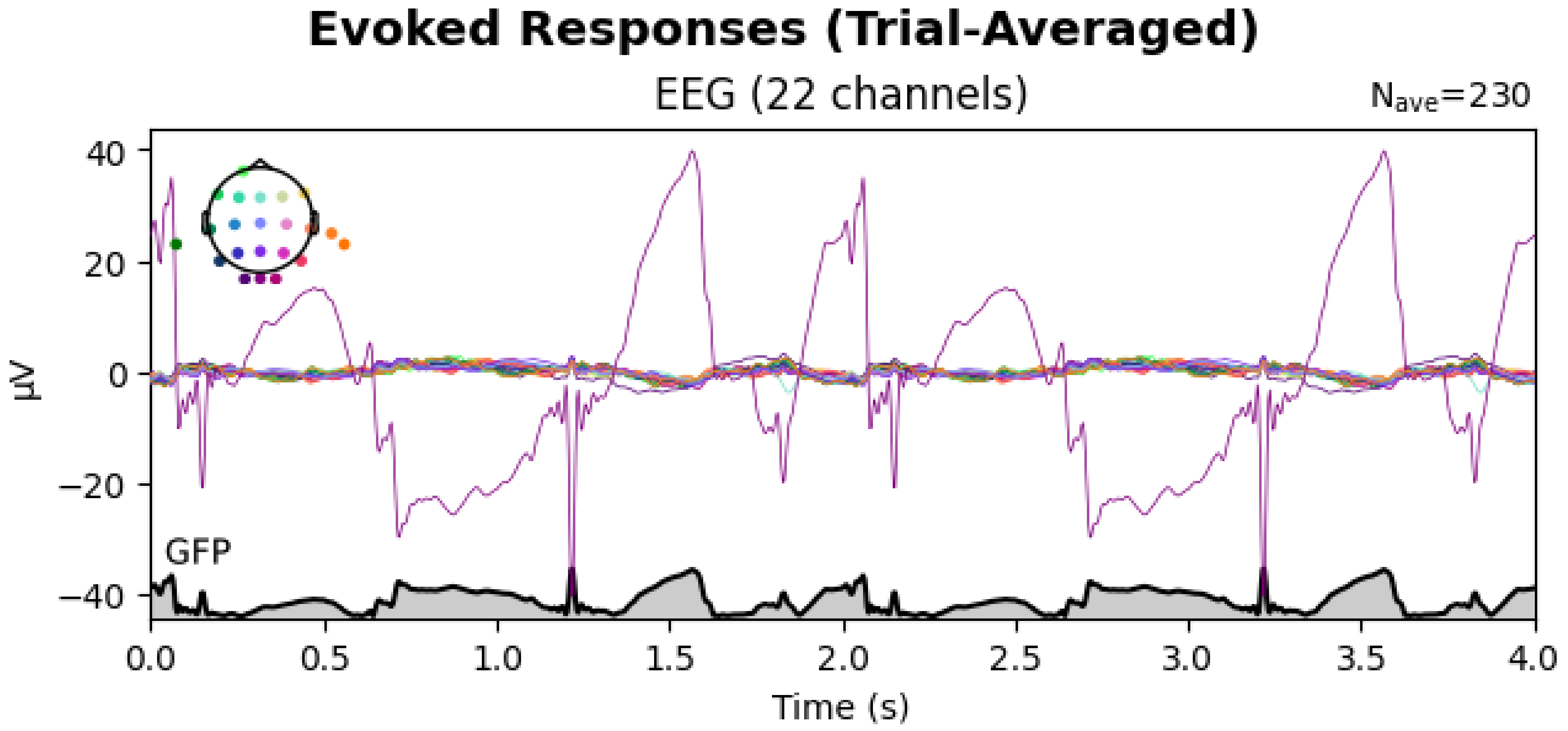

seizure



non-seizure

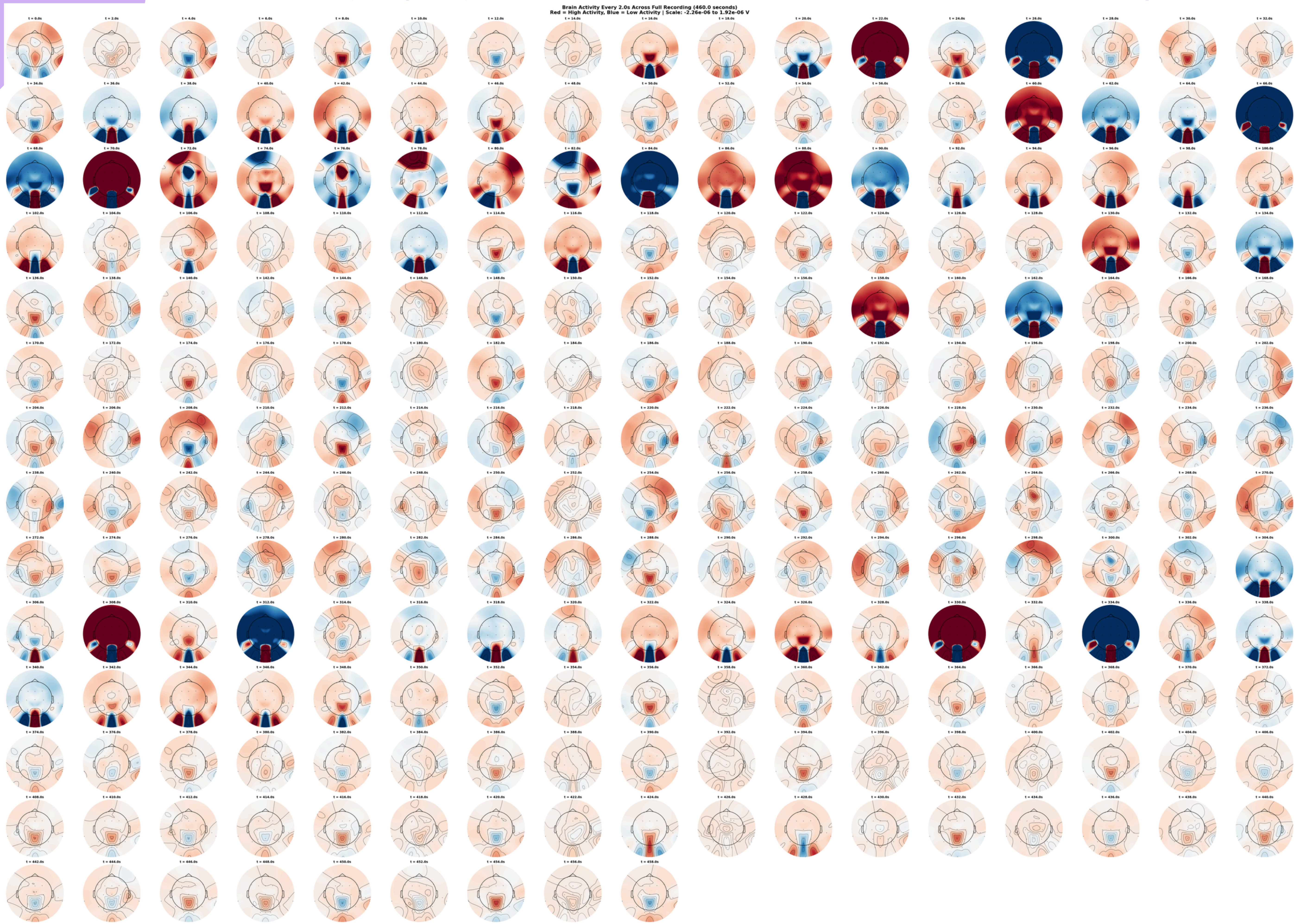



seizure



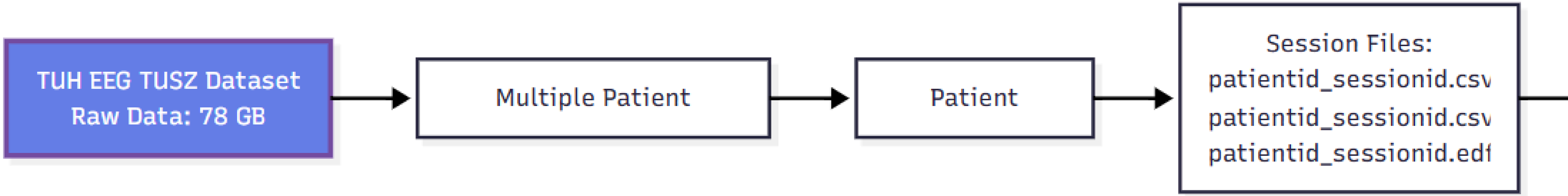
# non-seizure topographic maps over the whole recording

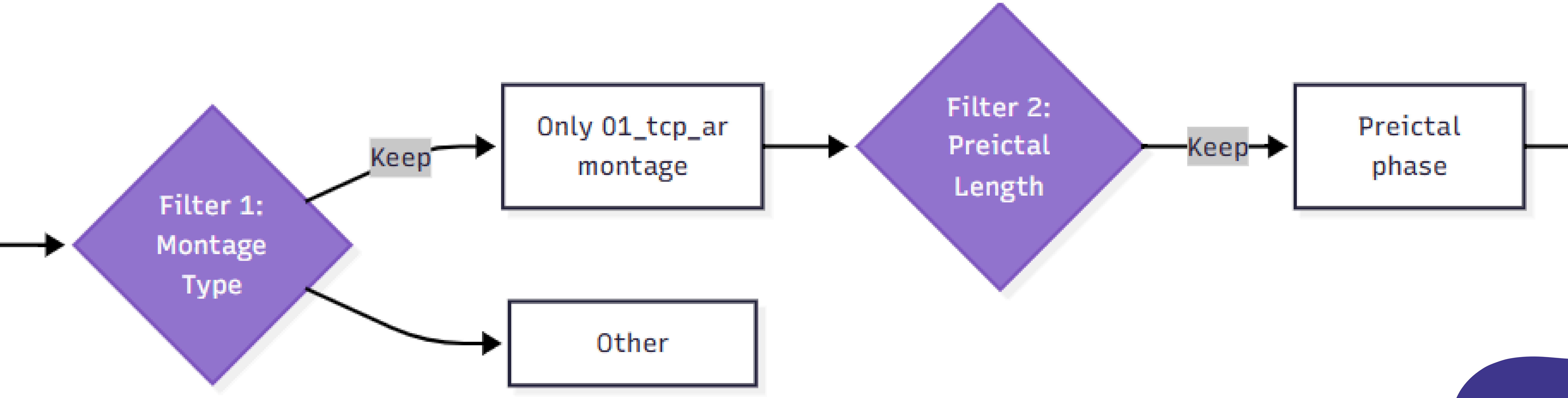
# seizure topographic maps over the whole recording

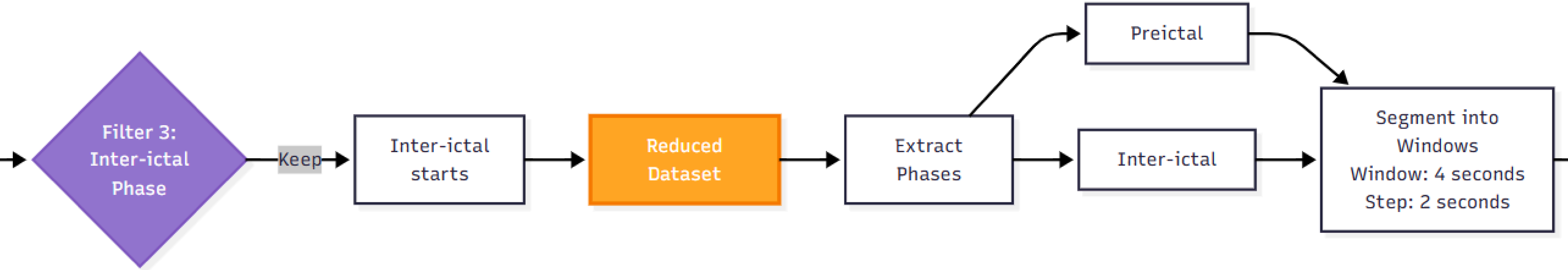


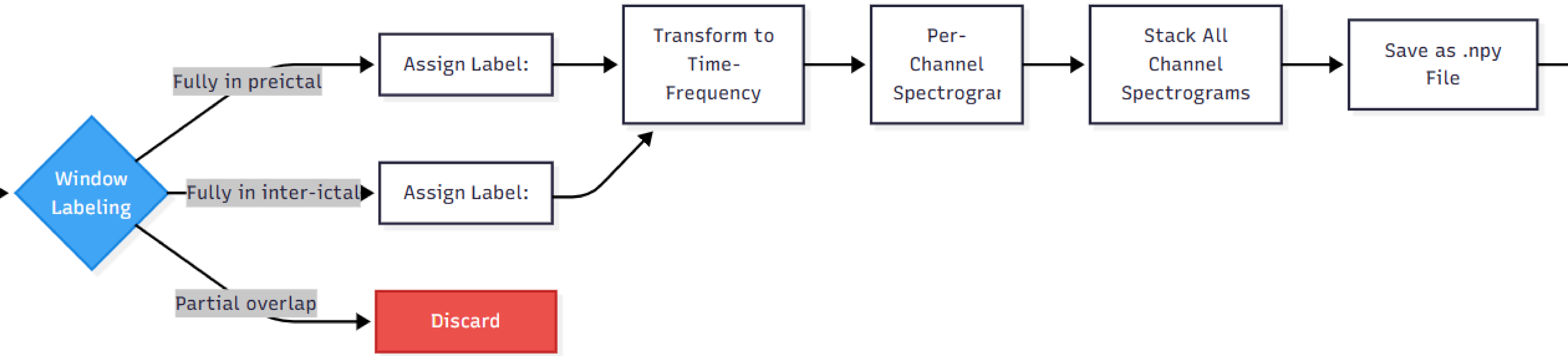
# DATA PREPROCESSING PIPELINE

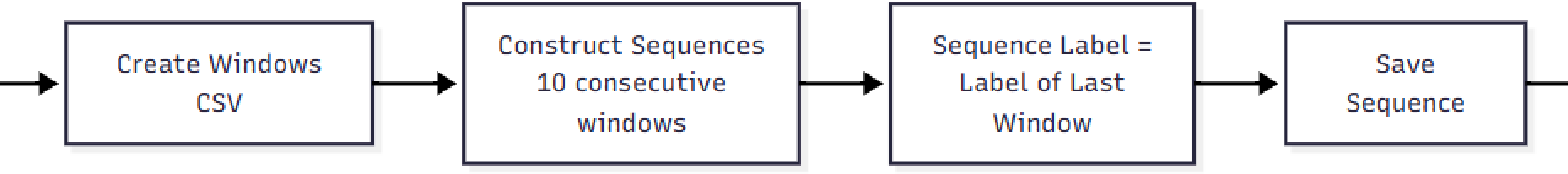
1. Filtering
2. Selection
3. Transformation
4. Construction

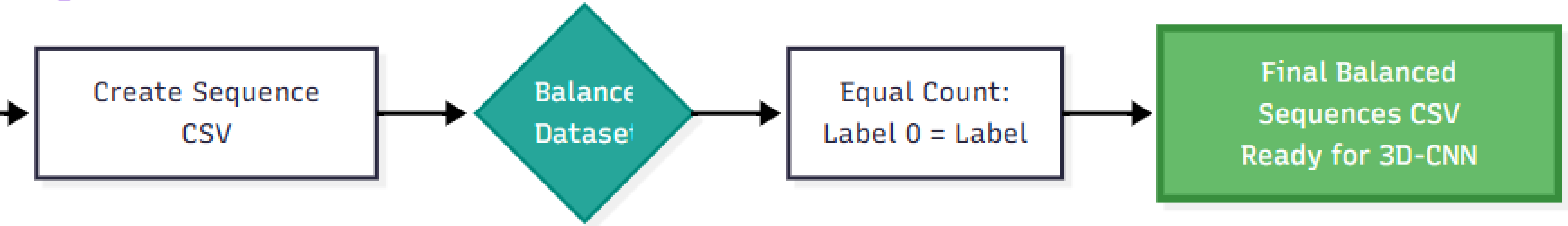

TUH EEG TUSZ Dataset
Raw Data: 78 GB
Multiple Patient
Patient
Session Files:
patientid_sessionid.csv
patientid_sessionid.csv
patientid_sessionid.edf


Filter 1:
Montage
Type
Keep
Only 01_tcp_ar
montage
Other
Filter 2:
Preictal
Length
Keep
Preictal
phase

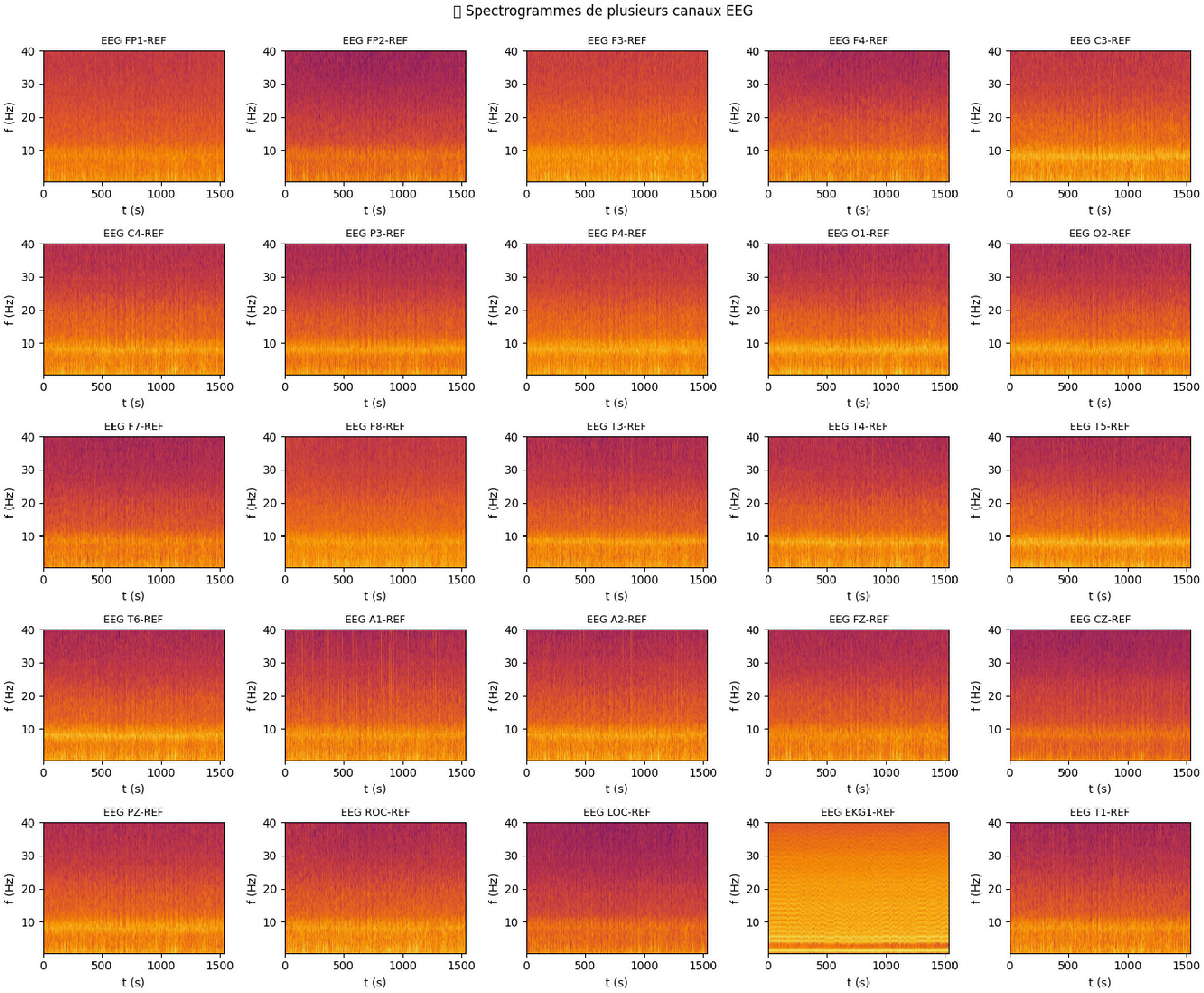

Filter 3: Inter-ictal Phase
Keep
Inter-ictal starts
Reduced Dataset
Extract Phases
Preictal
Inter-ictal
Segment into Windows
Window: 4 seconds
Step: 2 seconds


Window Labeling
Fully in preictal
Fully in inter-ictal
Partial overlap
Assign Label:
Assign Label:
Discard
Transform to Time-Frequency
Per-Channel Spectrograr
Stack All Channel Spectrograms
Save as .npy File

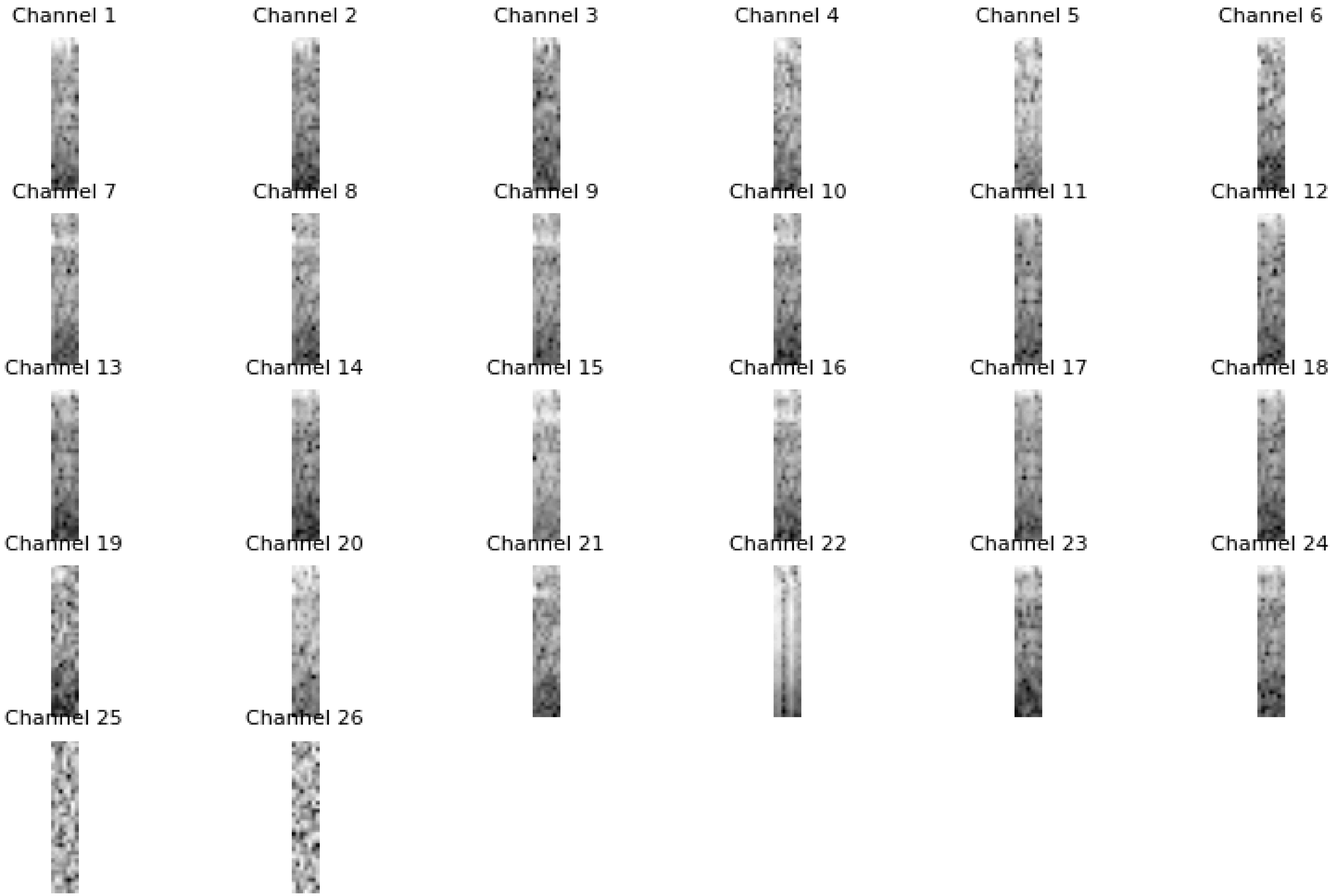

Create Windows CSV
Construct Sequences 10 consecutive windows
Sequence Label = Label of Last Window
Save Sequence


Create Sequence CSV
Balance Dataset
Equal Count: Label 0 = Label
Final Balanced Sequences CSV Ready for 3D-CNN

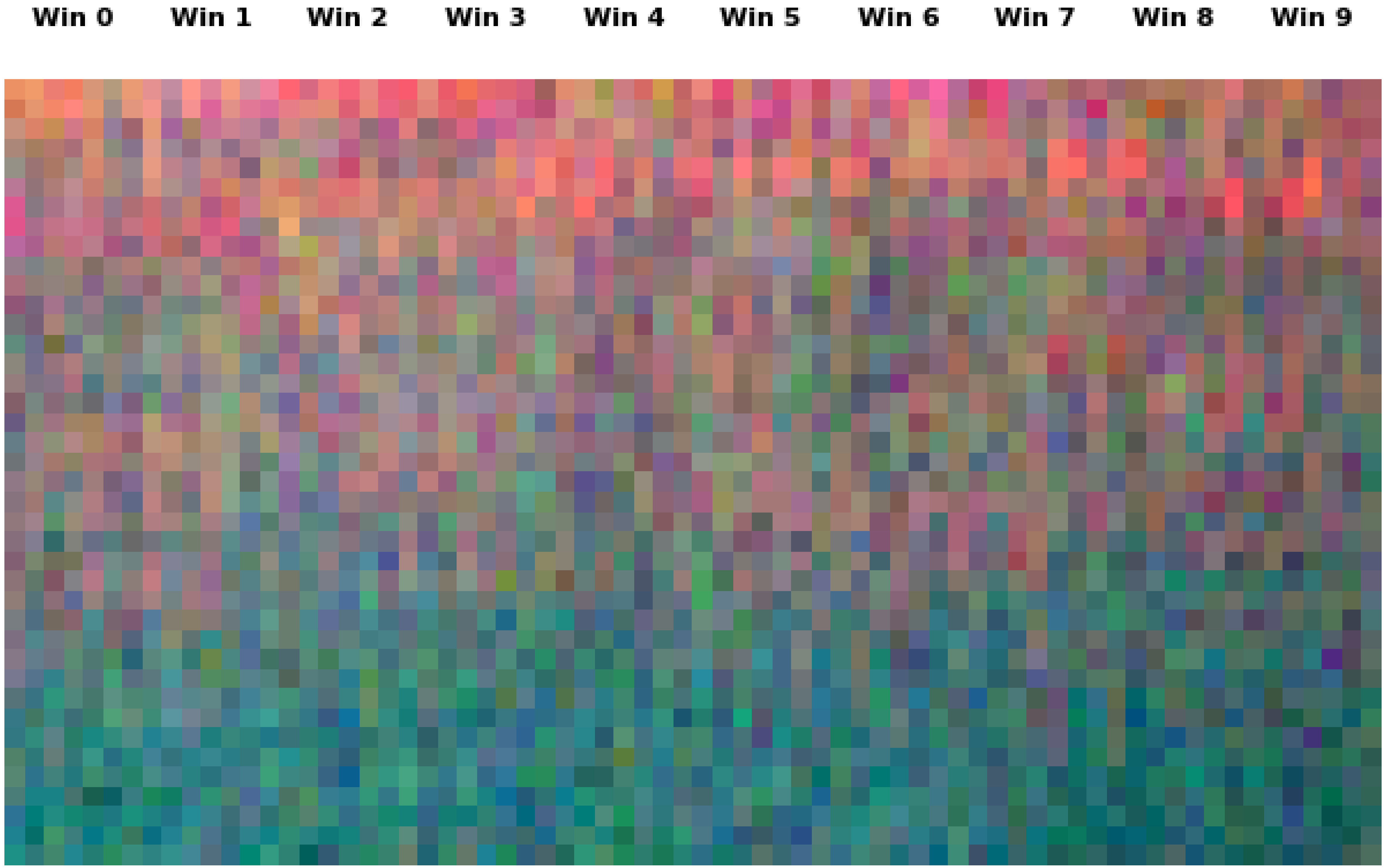
# spectrogram of an entire recording per channel



# window (4s) spectrogram (grayscale)



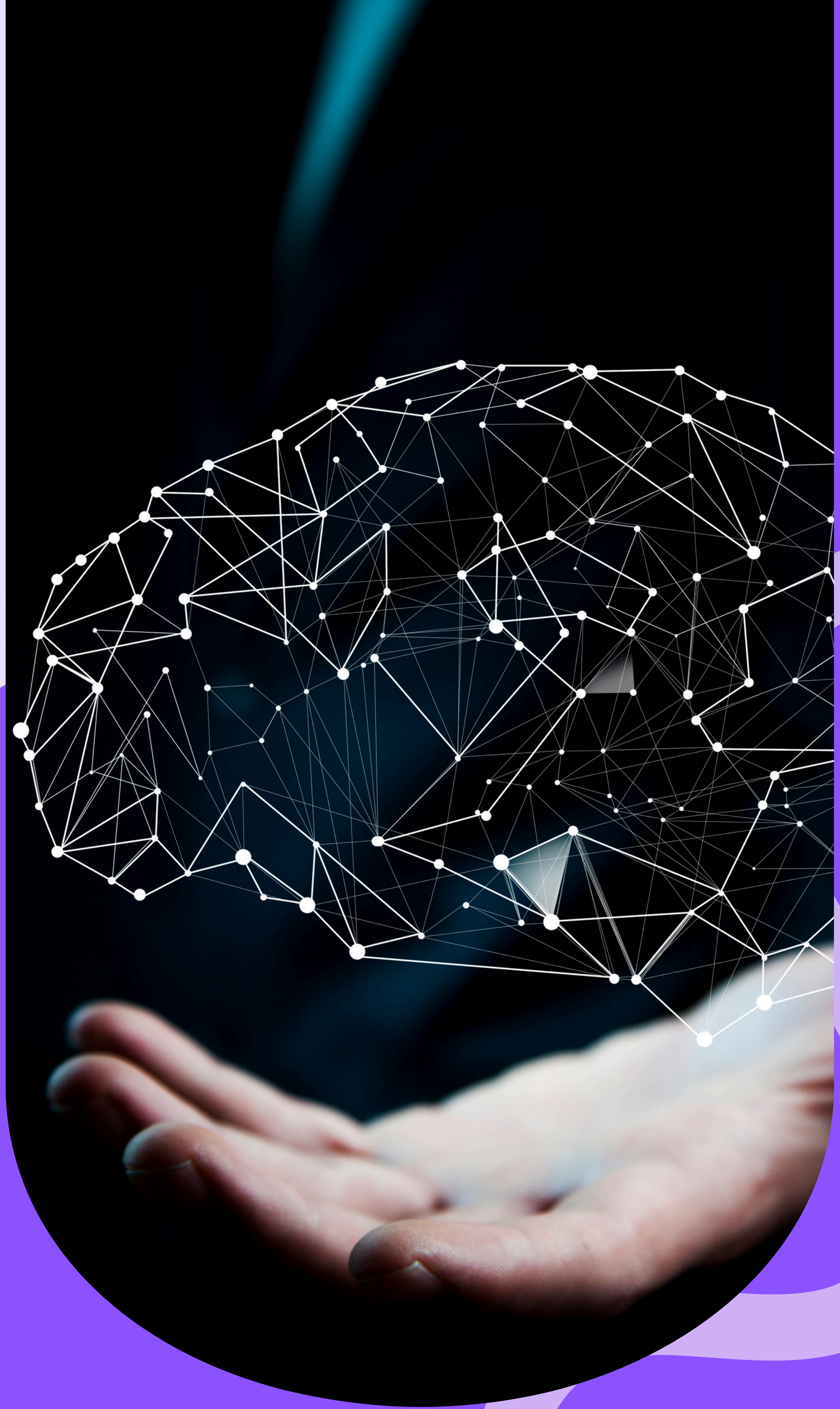
# sequence FOR VISUALIZATION PURPOSE

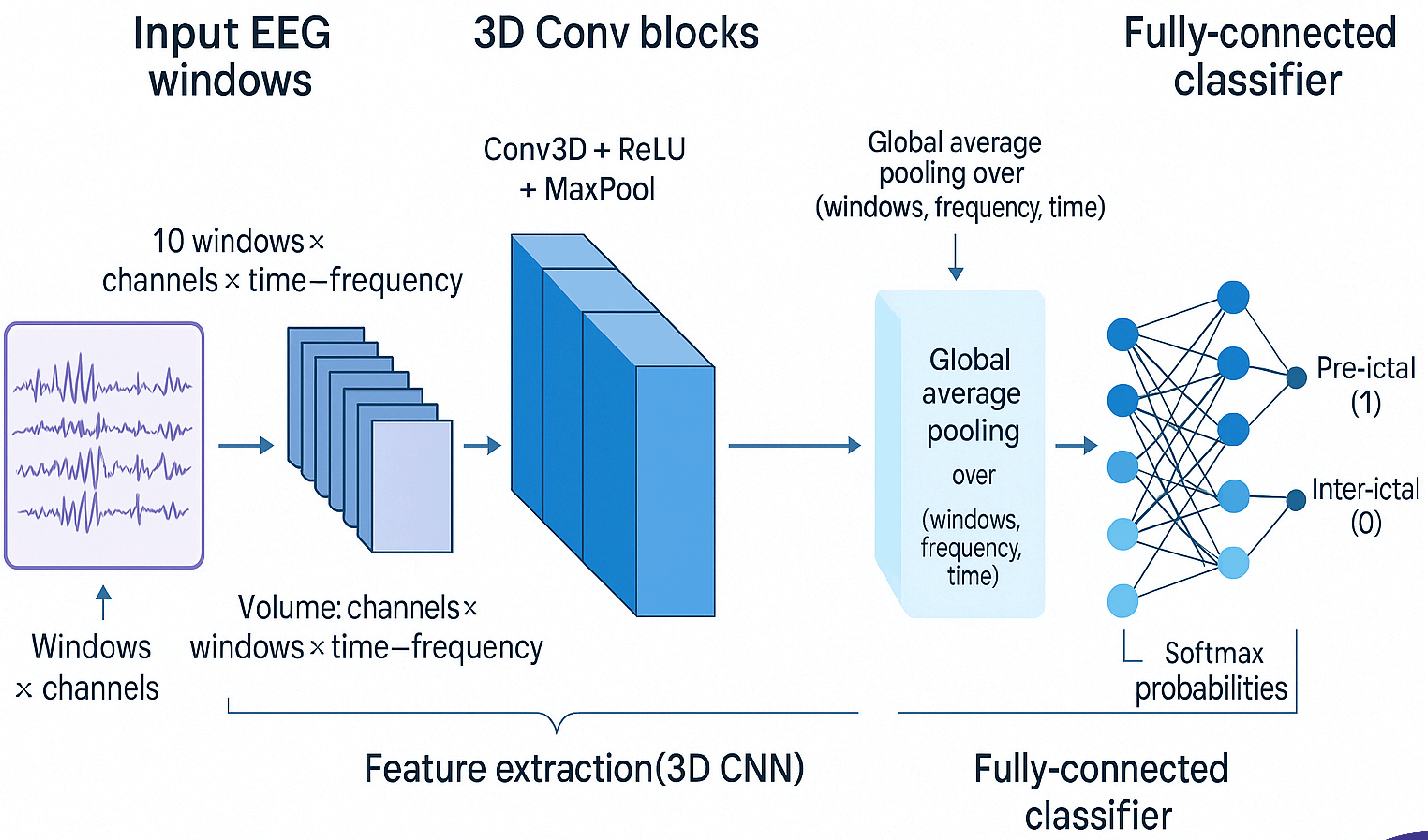



for the sake of visualization we applied PCA over the 26~29 channels of each window to end up with 3 channels to mimic RGB and we obtained the following image of our sequence

# Walkthroughs in Our CNN Model

- ARCHITECTURE
- RESULTS





Input EEG windows
3D Conv blocks
Fully-connected classifier
Conv3D + ReLU + MaxPool
Global average pooling over (windows, frequency, time)
10 windows × channels × time–frequency
Global average pooling over (windows, frequency, time)
Pre-ictal (1)
Inter-ictal (0)
Windows × channels
Volume: channels × windows × time–frequency
Softmax probabilities
Feature extraction(3D CNN)
Fully-connected classifier

# displaying some statistics about our final dataset

```
prepared\all\train_sequences_index_balanced.csv
CSV utilisé : prepared\all\train_sequences_index_balanced.csv
                                            path  label   patient    session   \
0  all/sequences/aaaaarjc_s007_t002_seq0083.npy      0  aaaaarjc  s007_2015
1  all/sequences/aaaaaqnr_s001_t000_seq0080.npy      0  aaaaaqnr  s001_2014
2  all/sequences/aaaaaqtw_s002_t011_seq0037.npy      1  aaaaaqtw  s002_2014
3  all/sequences/aaaaanme_s010_t010_seq0105.npy      0  aaaaanme  s010_2014
4  all/sequences/aaaaaqvx_s002_t001_seq0199.npy      0  aaaaaqvx  s002_2015

            recording  last_win_center_s
0  aaaaarjc_s007_t002              186.0
1  aaaaaqnr_s001_t000              180.0
2  aaaaaqtw_s002_t011              156.0
3  aaaaanme_s010_t010              572.0
4  aaaaaqvx_s002_t001             2658.0
Taille CSV : 9546
label
0    4773
1    4773
Name: count, dtype: int64
```

# slicing data into : train , eavl , test

```
Taille train : 4787
Taille val   : 2274
Taille test  : 2485

Distribution des labels (train)
label
0    2658
1    2129
Name: count, dtype: int64

Distribution des labels (val)
label
1    1156
0    1118
Name: count, dtype: int64

Distribution des labels (test)
label
1    1488
0     997
Name: count, dtype: int64
```

# applied grid search over 8 different configs to find the best hyperparameters

```
  Epoch 4/5 - Train loss: 0.2195, acc: 0.916 | Val loss: 3.1655, acc: 0.478
  Epoch 5/5 - Train loss: 0.1665, acc: 0.936 | Val loss: 1.9949, acc: 0.487

===== Config 6/8 =====
{'base_filters': 32, 'dropout': 0.3, 'lr': 0.0003}
  Epoch 1/5 - Train loss: 0.5268, acc: 0.725 | Val loss: 1.4596, acc: 0.481
  Epoch 2/5 - Train loss: 0.3008, acc: 0.874 | Val loss: 1.5081, acc: 0.515
  Epoch 3/5 - Train loss: 0.1846, acc: 0.930 | Val loss: 1.9154, acc: 0.488
  Epoch 4/5 - Train loss: 0.1489, acc: 0.945 | Val loss: 4.1871, acc: 0.493
  Epoch 5/5 - Train loss: 0.1124, acc: 0.959 | Val loss: 2.6590, acc: 0.493

===== Config 7/8 =====
{'base_filters': 32, 'dropout': 0.5, 'lr': 0.001}
  Epoch 1/5 - Train loss: 0.5725, acc: 0.688 | Val loss: 0.9749, acc: 0.455
  Epoch 2/5 - Train loss: 0.3827, acc: 0.830 | Val loss: 2.2206, acc: 0.475
  Epoch 3/5 - Train loss: 0.2774, acc: 0.884 | Val loss: 1.3930, acc: 0.469
  Epoch 4/5 - Train loss: 0.2104, acc: 0.915 | Val loss: 1.4664, acc: 0.469
  Epoch 5/5 - Train loss: 0.1715, acc: 0.933 | Val loss: 1.8539, acc: 0.493

===== Config 8/8 =====
{'base_filters': 32, 'dropout': 0.5, 'lr': 0.0003}
  Epoch 1/5 - Train loss: 0.5320, acc: 0.722 | Val loss: 1.6043, acc: 0.500
  Epoch 2/5 - Train loss: 0.3233, acc: 0.867 | Val loss: 1.2976, acc: 0.530
  Epoch 3/5 - Train loss: 0.2183, acc: 0.914 | Val loss: 1.7081, acc: 0.505
  Epoch 4/5 - Train loss: 0.1654, acc: 0.938 | Val loss: 2.4702, acc: 0.476
  Epoch 5/5 - Train loss: 0.1318, acc: 0.952 | Val loss: 3.1573, acc: 0.505

===== Meilleure config trouvée =====
{'base_filters': 16, 'dropout': 0.3, 'lr': 0.0003}
Best val acc : 0.5224274406332454
```

## Train

```
Epoch 1/20 - Train loss: 0.5473, acc: 0.700 | Val loss: 1.1719, acc: 0.504
  -> Nouveau meilleur modèle sauvegardé.
Epoch 2/20 - Train loss: 0.3312, acc: 0.862 | Val loss: 1.3441, acc: 0.503
Epoch 3/20 - Train loss: 0.2117, acc: 0.915 | Val loss: 1.4210, acc: 0.490
Epoch 4/20 - Train loss: 0.1446, acc: 0.949 | Val loss: 1.5470, acc: 0.504
Epoch 5/20 - Train loss: 0.1212, acc: 0.957 | Val loss: 2.8695, acc: 0.494
Epoch 6/20 - Train loss: 0.1040, acc: 0.962 | Val loss: 1.6452, acc: 0.502
  -> Early stopping déclenché.

Entraînement terminé.
Meilleur modèle enregistré dans : models/best_cnn3d_tuh.pth
```

## Test

```
Confusion matrix :
 [[701 296]
 [871 617]]

Classification report :
              precision    recall  f1-score   support

           0      0.446     0.703     0.546       997
           1      0.676     0.415     0.514      1488

    accuracy                          0.530      2485
   macro avg      0.561     0.559     0.530      2485
weighted avg      0.584     0.530     0.527      2485
```

# THANK YOU FOR YOUR ATTENTION


Presented by:

OUDAY MESSAADI

OUSSEMA CHERNI

## Contact:

EMAIL ADDRESS

ouday.messaadi@etudiant-enit.utm.tn

oussema.cherni@etudiant-enit.utm.tn