Towards Understanding Plant Reactions to Human Movement

Machine Learning Analysis of the Electrical Signal of Plant Species Exposed to Eurythmy Letters

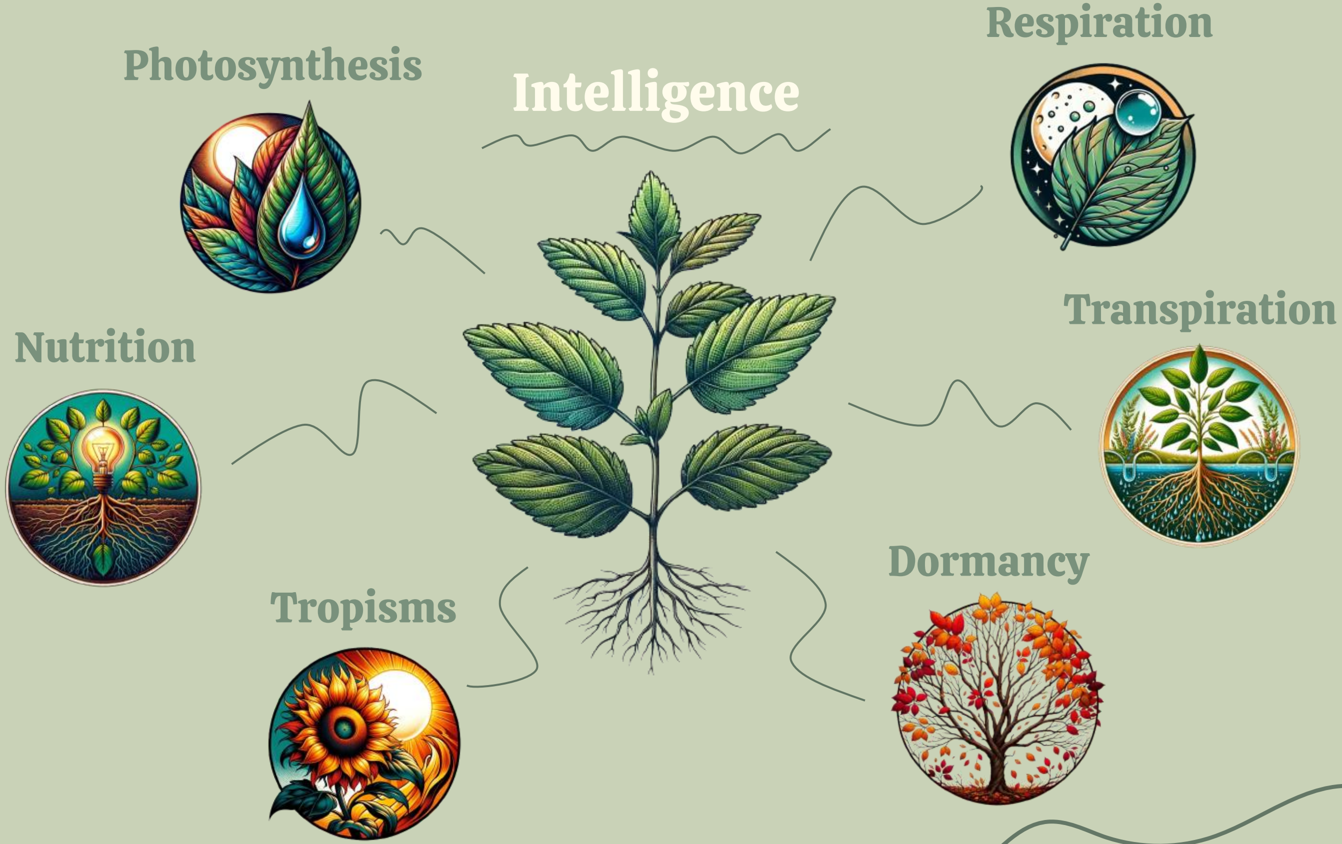
Table of contents

01

Introduction

(Related Work)

What does a plant do?

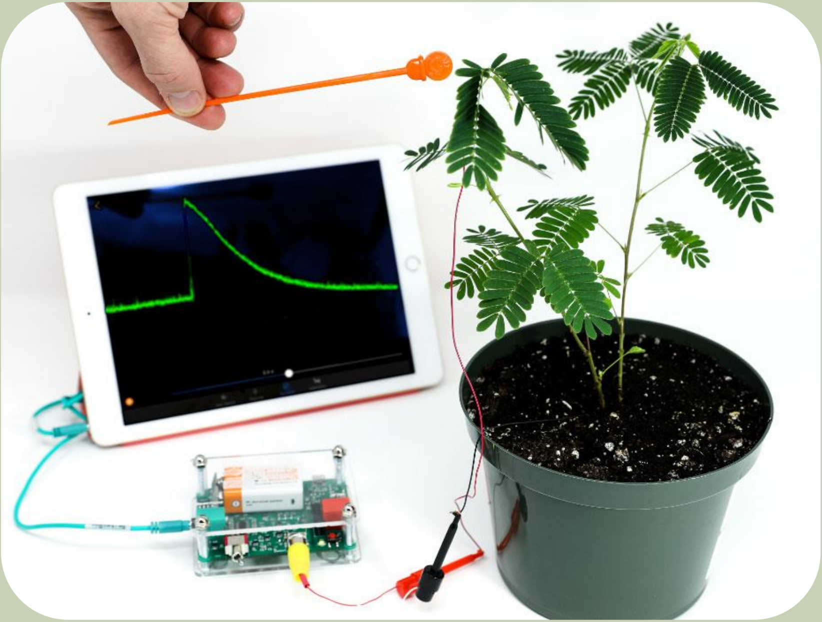


Electrophisiology

Electrical Responses

Sanderson, 1873
Pickard, 1973
Marzullo, 2012

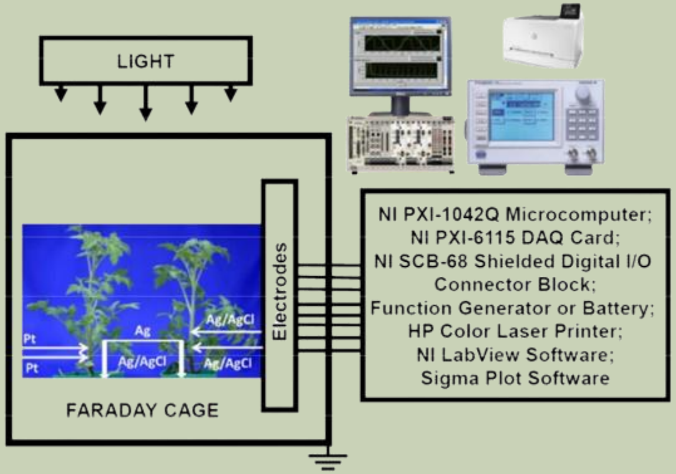
Plant Spikerbox



Communication

Volkov et al, 2019

1. Volatile Organic Compounds (chemicals)
2. Mycorrhizal Networks (fungi)
3. Rhizosphere (chemical)
4. Natural Grafting (merge roots)
5. Electrostatic or Electromagnetic
6. Acoustic
7. **Electrical Signal through the Soil**



Light Stimulus

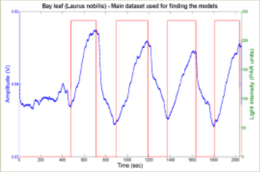
Prediction from Electrophysiological Response

Chatterjee et al, 2014

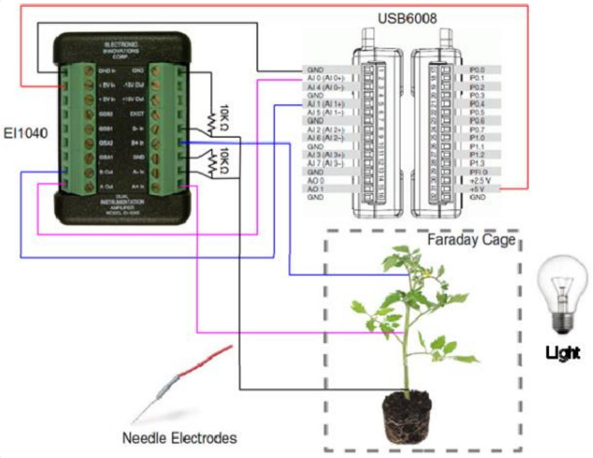
bay leaf

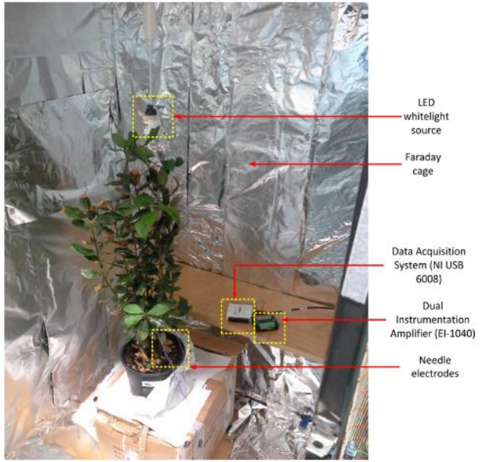
cucumber

zanzibar



Photon Flux Density

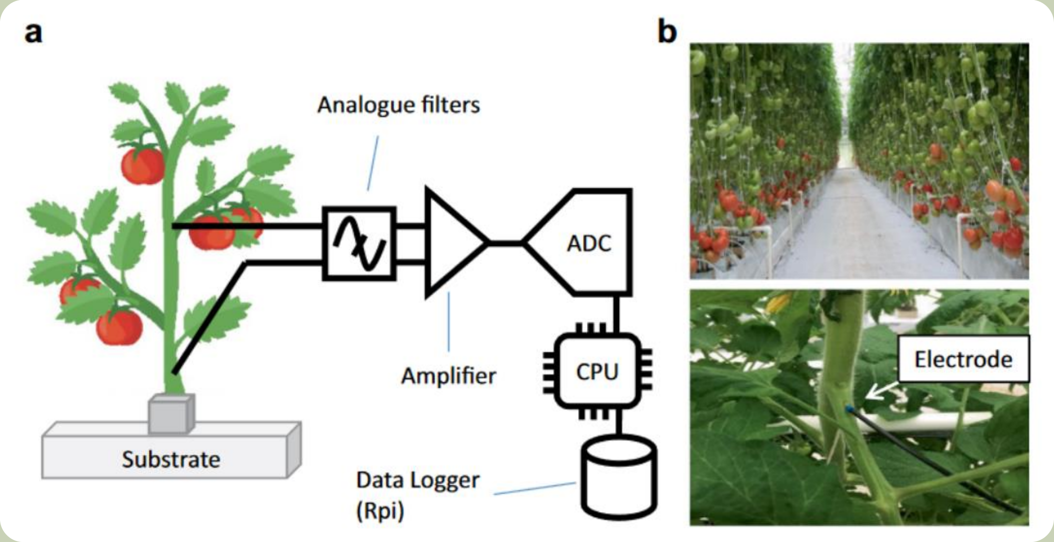




Water status

By electrophysiological assessment

Tran et al, Nature 2019



Monitoring

2 weeks

Output

Day/Night
Water Deficit

Acuracy

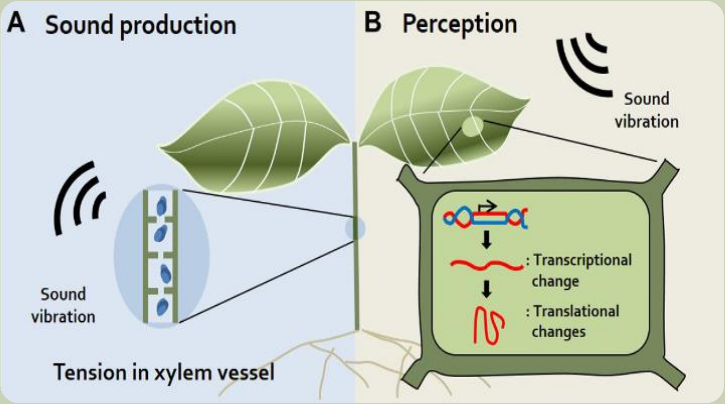
0.94
0.98

Sound

Production

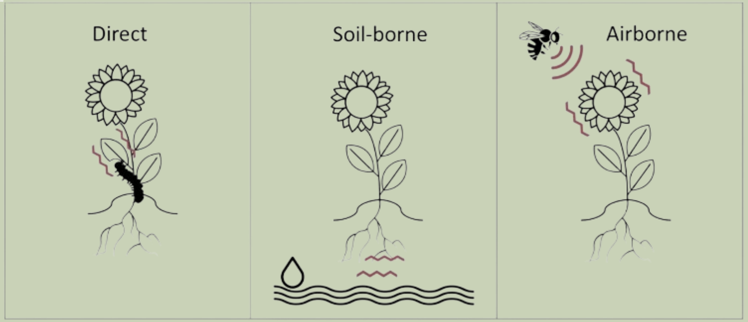
Jung et al, 2018

- Spontaneous sounds
- Gas bubbles in xylem vessels
- Audible and ultrasonic (20-105kHz)

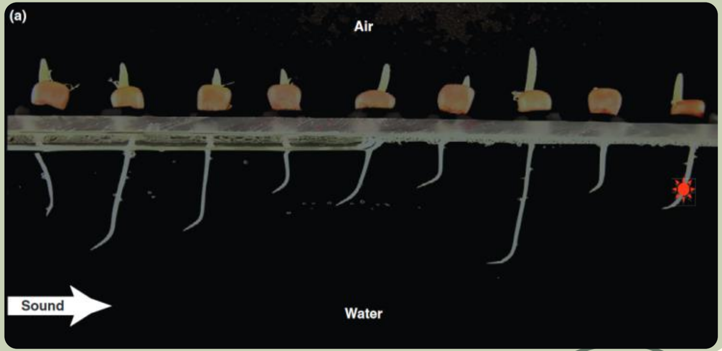


Perception

Khait et al, 2019

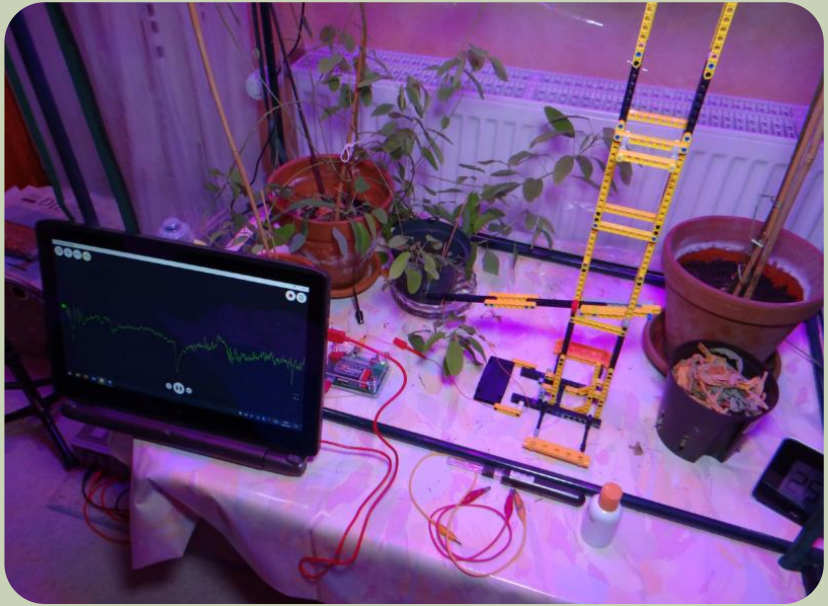


Gagliano et al, 2012



Music

Peter & Gloor, 2021



Location
Indoor greenhouse

Species
Dancing plant

Output
Control | 200 Hz | 600 Hz

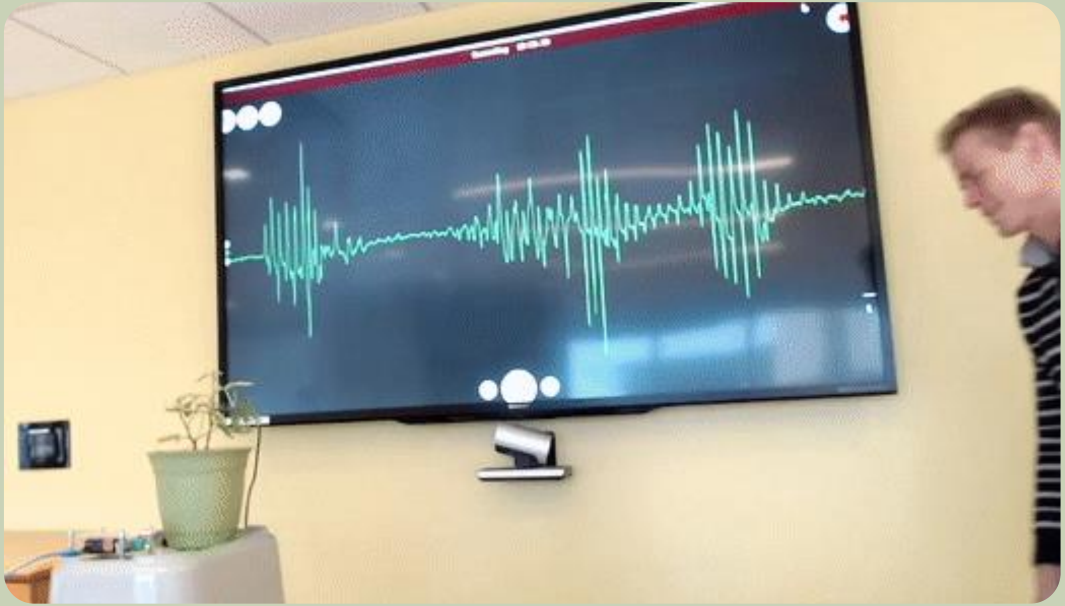
Accuracy
0.72

Mood
Person Detector
Oezkaya & Gloor, 2020

Species
Mimosa Pudica

Output
6 people
+/- mood

Accuracy
0.66
0.86



Eurythmy

- **Expressive movement art**
- Originated by **Rudolf Steiner**
- Used in education, anthroposophic medicine, and in **Biodynamic agriculture**
- The melody is conveyed through expressing the **arm gestures** of the actual letters

Research Questions

01 Do plants react to eurythmy?

02 Do plants react differently between different eurythmy gestures?

03 Do the electrical signals of different plant species contain unique characteristics?

02

Methodology

Methodology

01 Experiments
Explanation of the data collection process

02 Preparation
Labeling, trimming and normalization.

03 Featurization
Extraction of wave characteristics

04 Modeling
Selection, modeling and fine-tuning

Methodology

01

Experiments

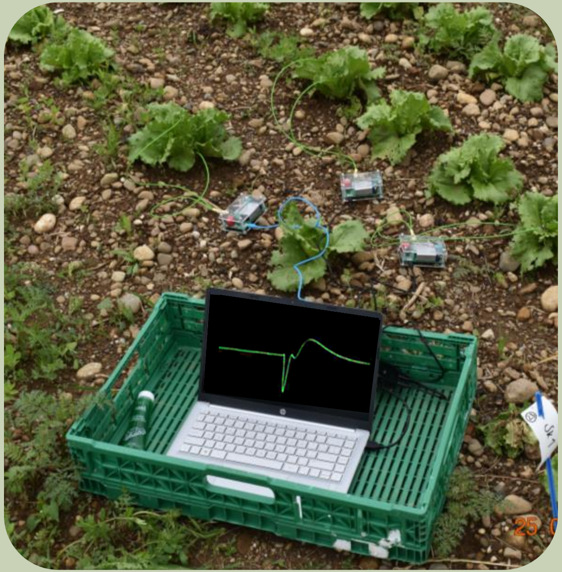
How to Measure

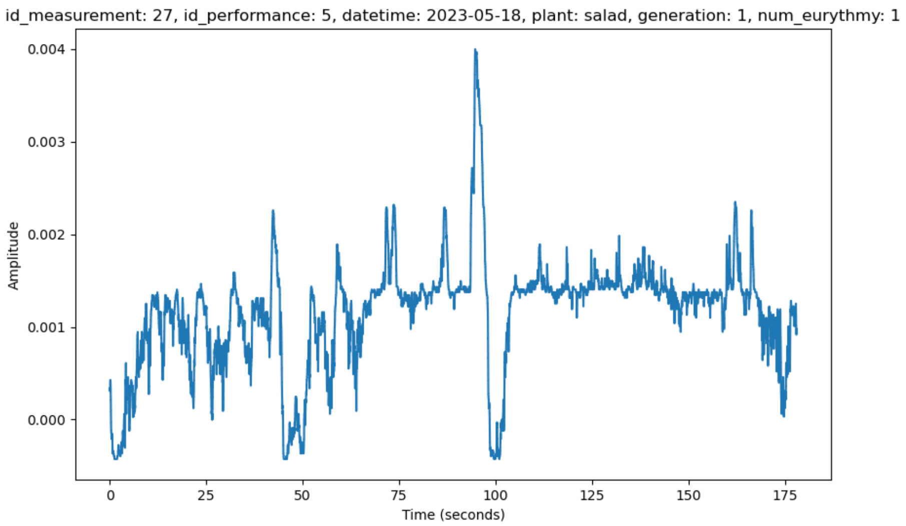
Extracellular Recording

- Surface measurements (EEG)
- Total sum of bioelectrical activity in large groups of cells
- Placement of electrodes on the plant surface and soil

https://backyardbrains.com/products/plantspikerbox

Electrical Plant Recording





Measurements

Salad

A-G-D

Tomato

A-G-D-O

Basil

A-G-D-L

Eurythmy Gestures

(4x times each)

Procedure

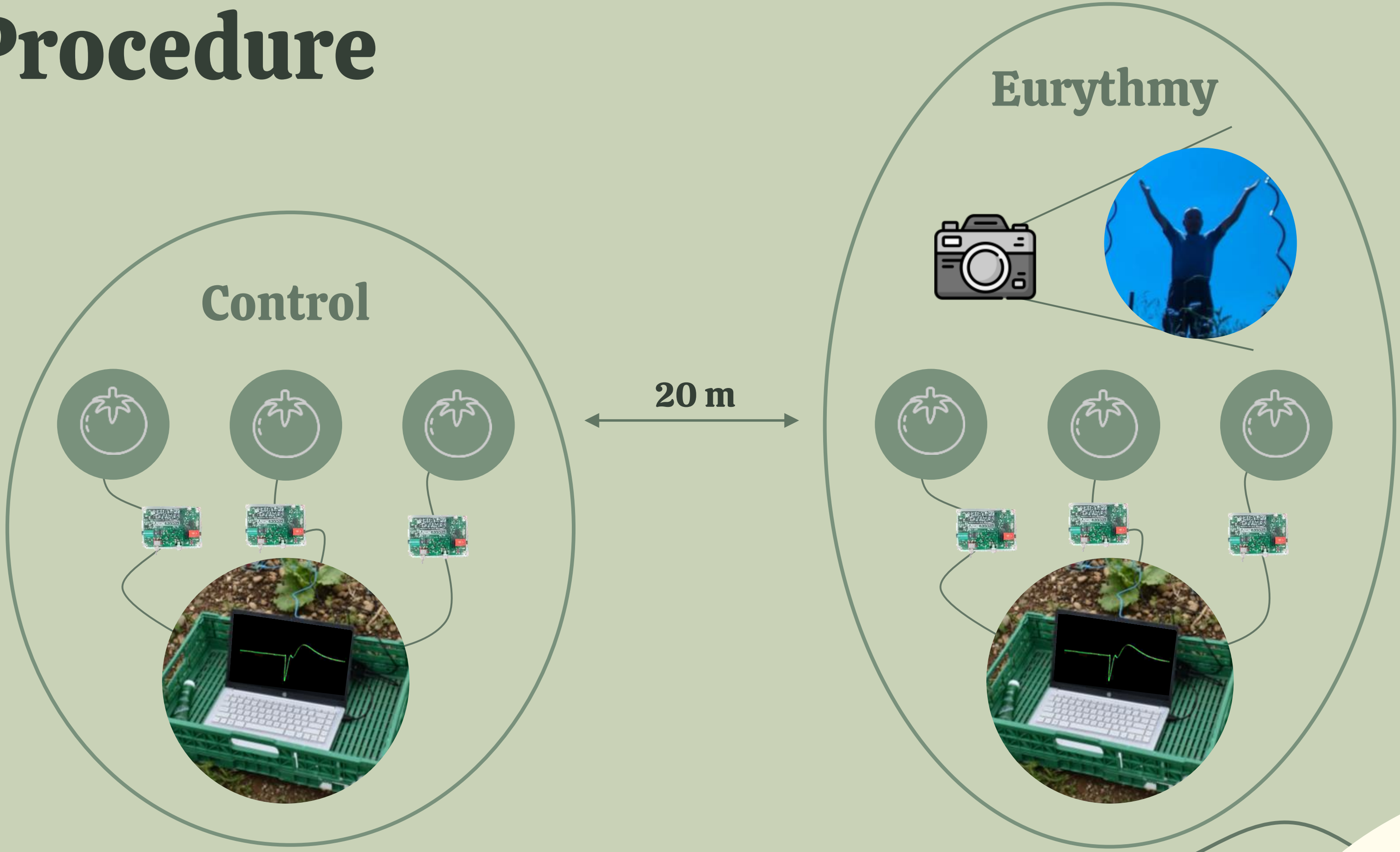
Control

The tension of 3 plants far away from the expert but in the same orchard is measured

Eurythmy

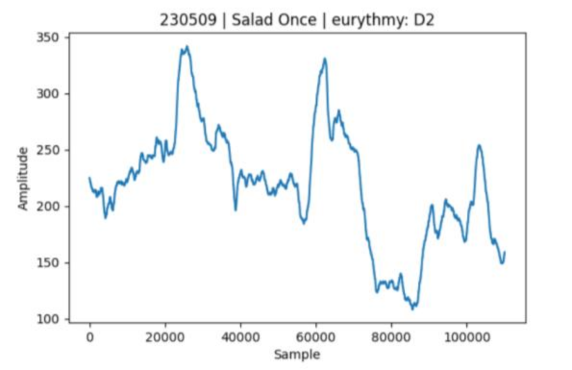
The voltage of 3 plants is measured while an expert performs the eurythmy gestures on them

Procedure



Files

Wav



The Wav files are datapoints and represent the plant voltage over time

625 files : 42 hours

Mp4

The Mp4 files help us to detect the gesture and timing of eurythmy

107 files : 7 hours

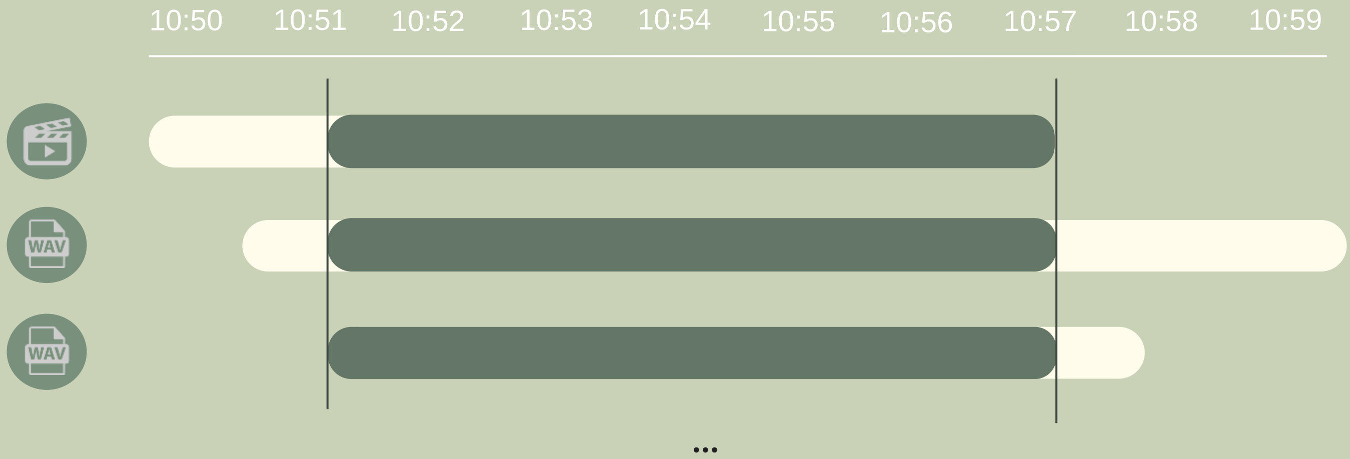
Methodology

02

Preparation

Synchronization

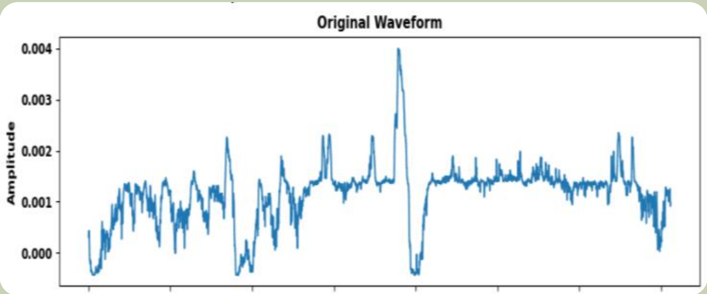
Intersection



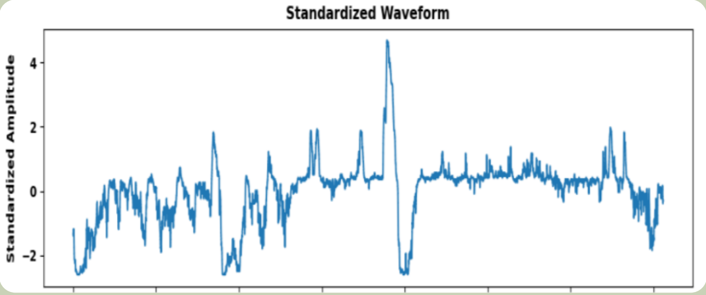
Standardization

Z-Score Normalization

```python
def standardize_wave_zscore(waveform):
    mean = np.mean(waveform)
    std_dev = np.std(waveform)
    return (waveform - mean) / std_dev
```



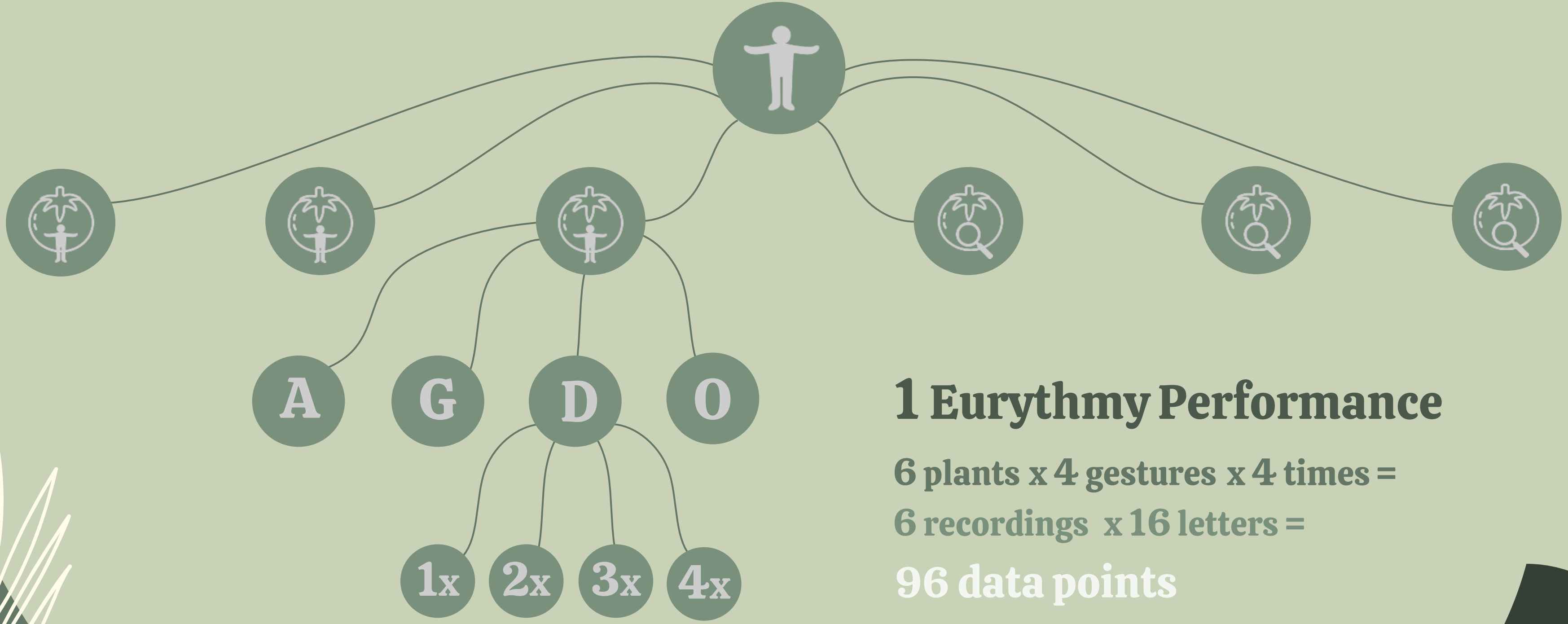
→



Different Gestures

#1	E	A1	G1	D1	A2	G2	D2	A3
#2	K	A1	G1	D1	A2	G2	D2	A3
		DP1	DP2	...				

Data Points



1 Eurythmy Performance

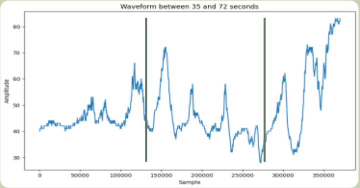
6 plants x 4 gestures x 4 times =

6 recordings x 16 letters =

96 data points

Trim

Labelling the video



+

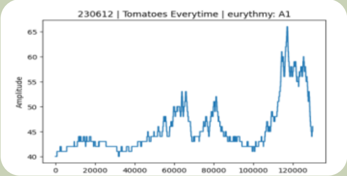
230612 TJ

File Edit View

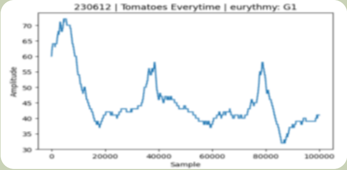
```
id, start, end
A1, 00:35,00:48
G1, 00:50,01:00
D1, 01:02,01:12
A2, 01:16,01:27
G2, 01:28,01:37
D2, 01:39,01:49
A3, 01:53,02:03
G3, 02:04,02:15
D3, 02:17,02:27
A4, 02:30,02:39
G4, 02:40,02:49
D4, 02:51,03:01
```

Data Point

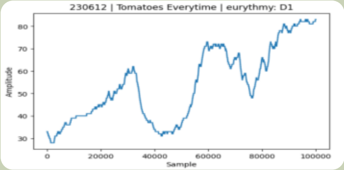
A



G



D



Data Count

Letter data points:

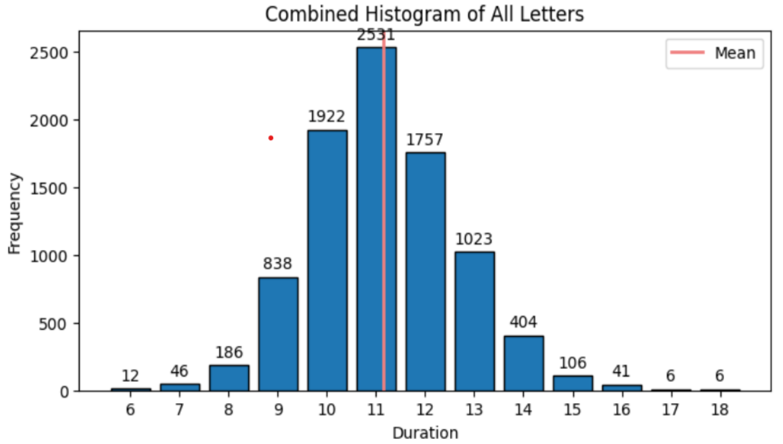
8878

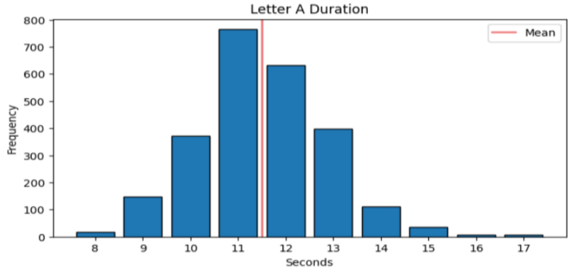
Control	Eurythmy
4395	4483

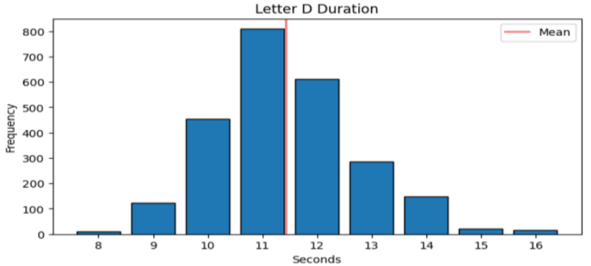
A	1261
G	1255
D	1255

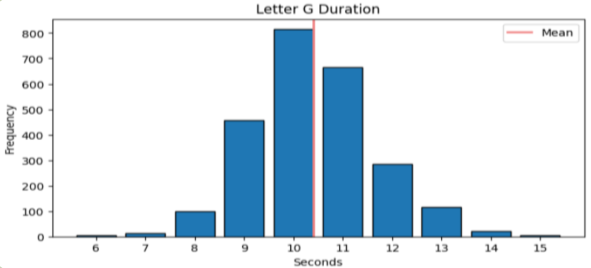
EDA

Letter Duration



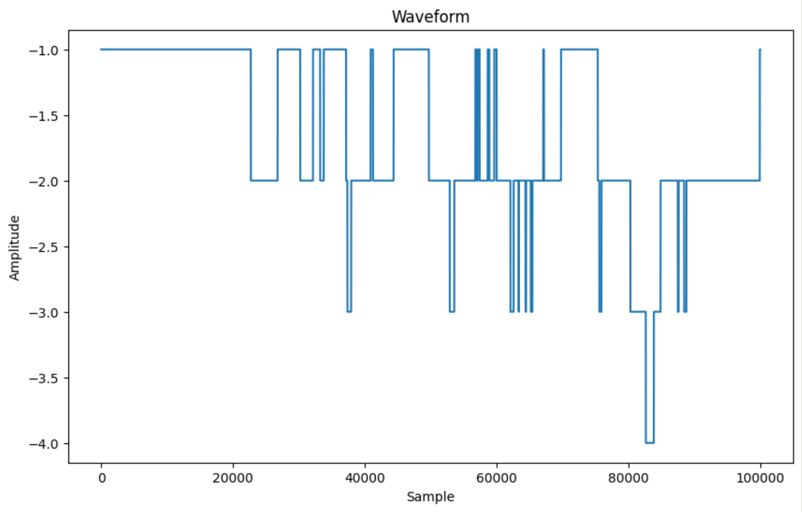


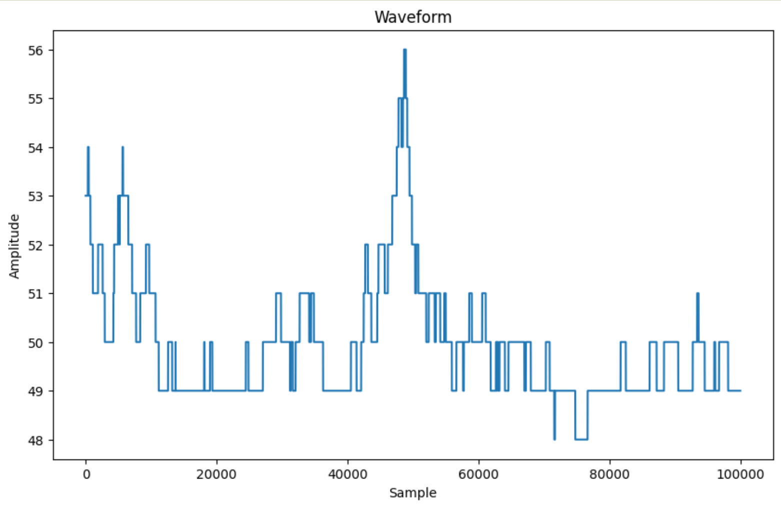




Outliers

How to detect them sistematically?

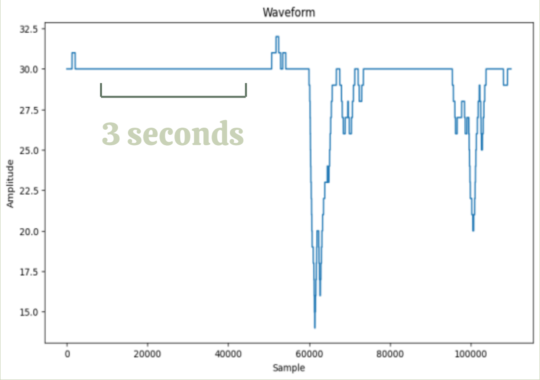




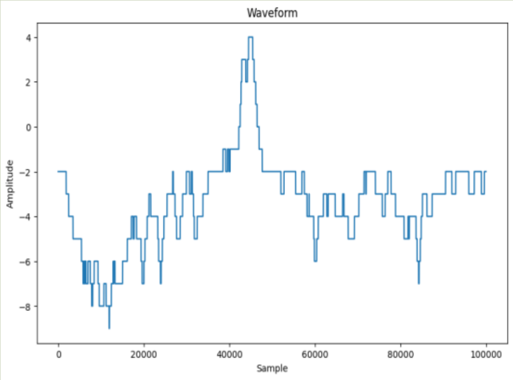
Outliers

Constant Value for X seconds

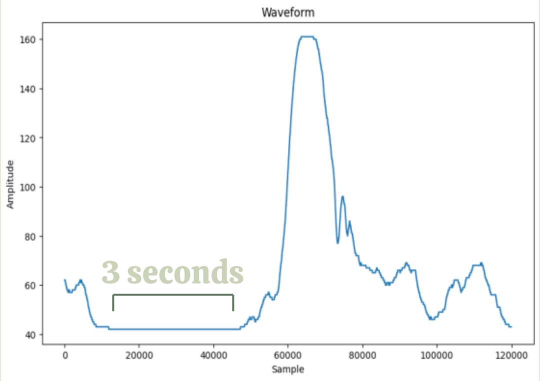
1st Approach



✓



✕

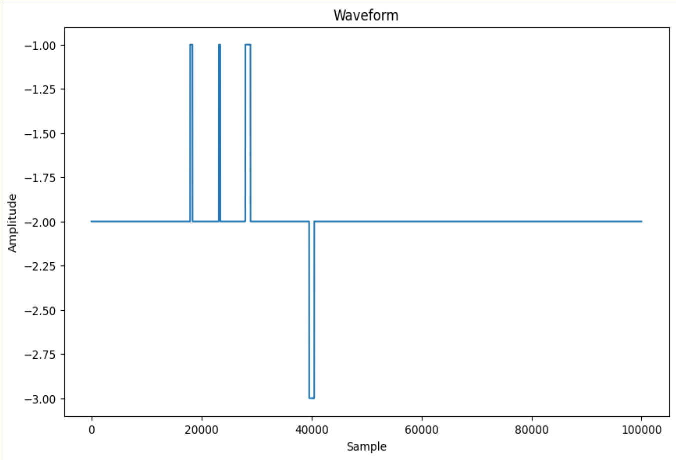


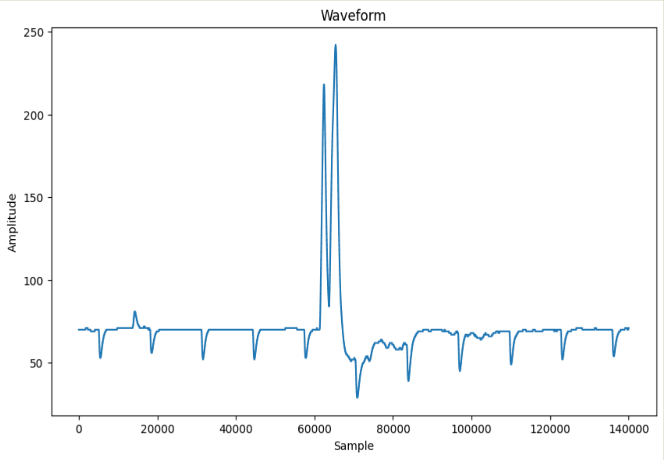
?

Outliers

Kurtosis treshold

2nd Approach





✓

✕

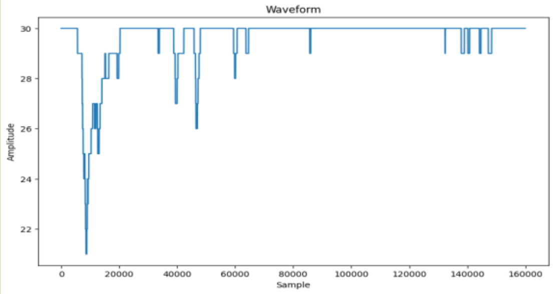
Outliers

Flatness Ratio

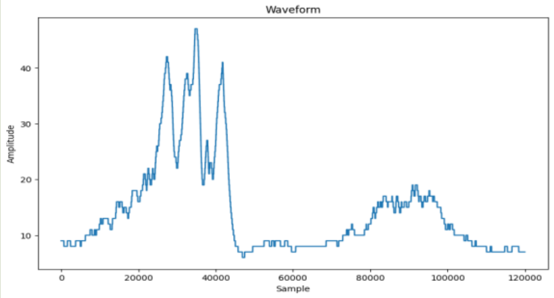
3rd Approach

20% dataset is > 50% Flatness Ratio

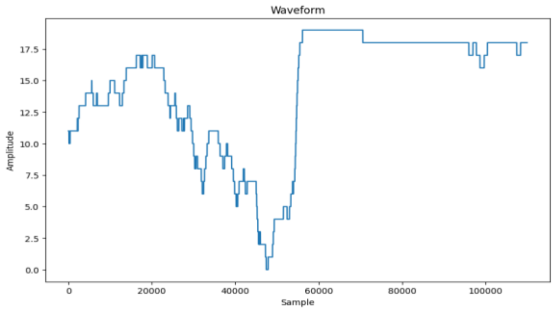
90%



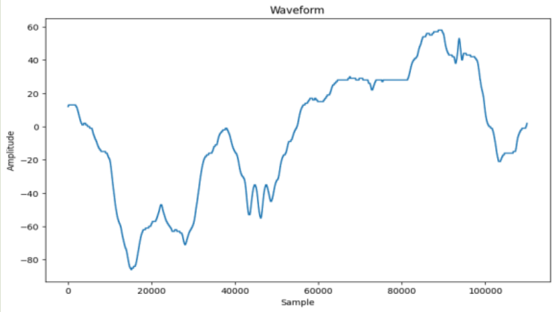
40%



70%



20%



Flatness Ratio

Definition

Formula

$$R = \frac{1}{n} \sum_{i=1}^{m} l_i \cdot 1(l_i > T)$$

Given an array A, the flatness ratio R is defined as the ratio between the count of consecutive elements exceeding a certain threshold T and the total number of elements:

Where

- $A = [a_1, a_2, \ldots, a_n]$: The array of n elements.
- $S = [s_1, s_2, \ldots, s_m]$: A sequence where each s_i represents a segment of consecutive occurrences of the same value in A.
- l_i: The length of the i^{th} segment s_i in S.
- T: The threshold for considering a segment's contribution to the flatness ratio.
- $1(\cdot)$: The indicator function, which is 1 if the condition inside the parentheses is true, and 0 otherwise.
- n: The total number of elements in A.
- m: The total number of segments in S.

Explanation

- $\sum_{i=1}^{m} l_i \cdot 1(l_i > T)$: This summation iterates through each segment s_i in S, multiplies its length l_i by 1 if $l_i > T$, and sums up these products.
- $\frac{1}{n}$: This term normalizes the sum by dividing it by the total number of elements n in A, yielding the flatness ratio R.

Flatness Ratio

Using Programming

Method

1. It iterates through the array, tracking sequences of identical values.

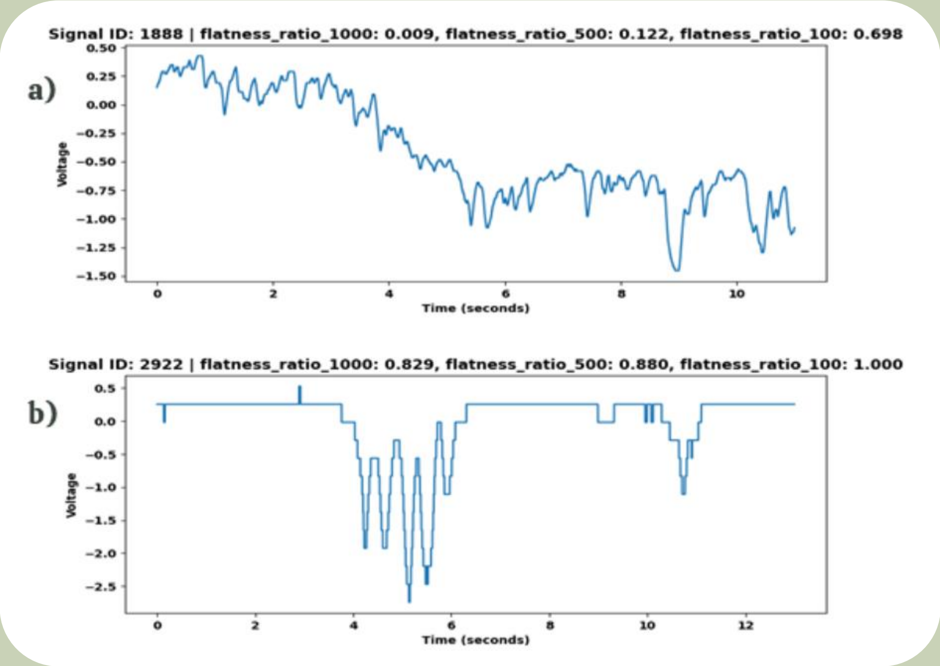
2. If a sequence length exceeds the threshold, it is added to the total 'flat' length.

3. The ratio of the total 'flat' length to the array's length is returned

```python
def extract_flatness_ratio(array, threshold):
    """
    Calculates the flatness ratio of an array.

    The flatness ratio is defined as the proportion of the array where
    consecutive values remain the same and exceed a specified threshold length.

    :param array: The input array to analyze.
    :param threshold: The minimum length of consecutive values to be considered 'flat'.
    :return: The flatness ratio of the array.

    The function works as follows:
    - It iterates through the array, tracking sequences of identical values.
    - If a sequence length exceeds the threshold, it is added to the total 'flat' length.
    - The ratio of the total 'flat' length to the array's length is returned.
    """
    current_value = None  # Tracks the current value being compared
    current_length = 0  # Tracks the length of the current sequence of identical values
    total_length = 0  # Accumulates the total length of all 'flat' sequences

    for value in array:
        if value == current_value:
            # If the current value matches the previous one, increment the sequence length
            current_length += 1
        else:
            # If the current value is different, reset the sequence
            current_value = value
            # If the previous sequence was long enough, add its length to the total
            if current_length > threshold:
                total_length += current_length
            current_length = 1  # Start a new sequence

    # Check the last sequence in the array
    if current_length > threshold:
        total_length += current_length

    # Calculate and return the flatness ratio
    return total_length / len(array)
```

Outlier Identification

Flatness Ratios



Flatness Ratio

In Datasets

FR	1000	500	100	
Alvaro	27%	44%	83%	+
Luis	24%	40%	81%	
Jakob	7%	10%	23%	
Saad	3%	7%	22%	-

Methodology

03

Featurization

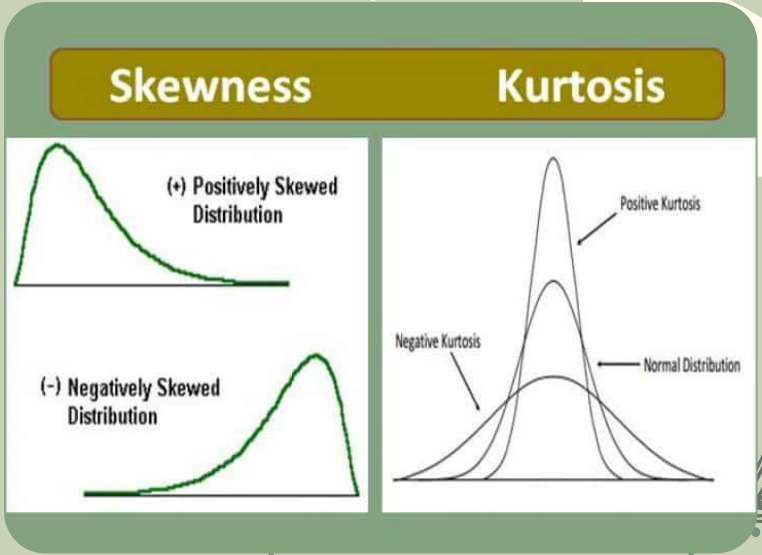
Featurization

Type of Features

- Statistical
- Temporal
- Frequency

Statistical Features

- Mean
- Variance
- Standard Deviation
- Interquartile Range
- Skewness
- Kurtosis
- Hjorth mobility
- Hjorth Complexity
- DFA

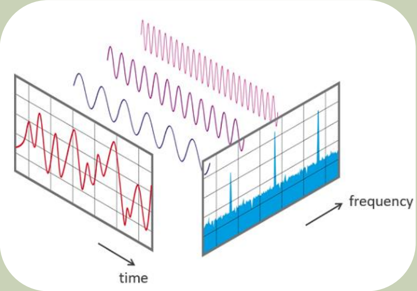


Temporal Features

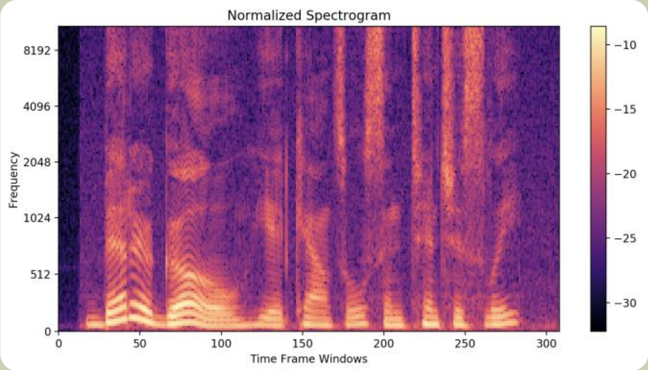
- Zero-Crossing Rate
- Root Mean Square Energy
- Slope Sign Changes Ratio

Frequency Processing

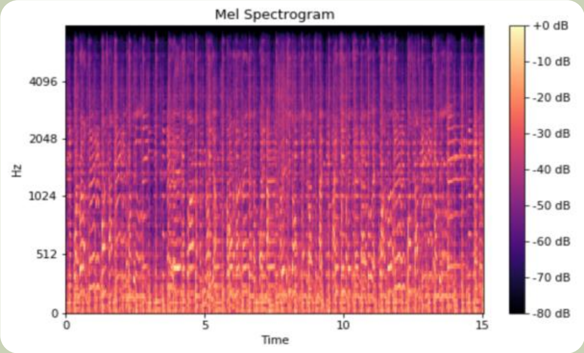
FFT



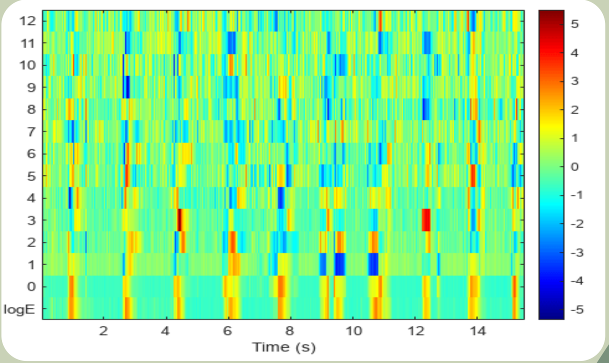
Spectrogram



Mel Spectrogram



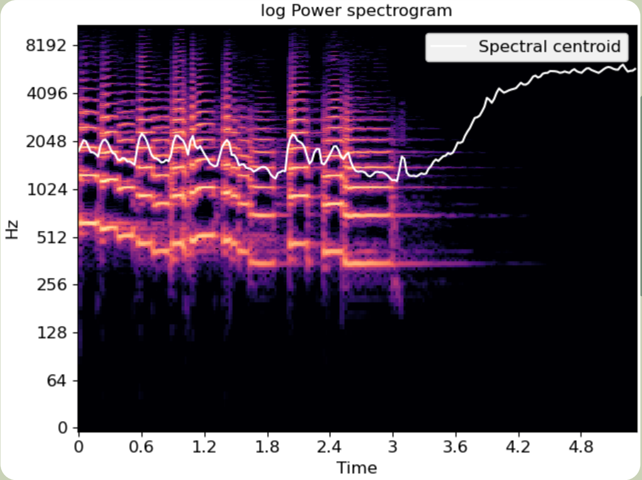
MFCC



Frequency Features

Mid-Term Features

- MFCCs
- Delta MFCCs
- Chromas
- Spectral Centroid
- Bandwith
- Energy Entropy



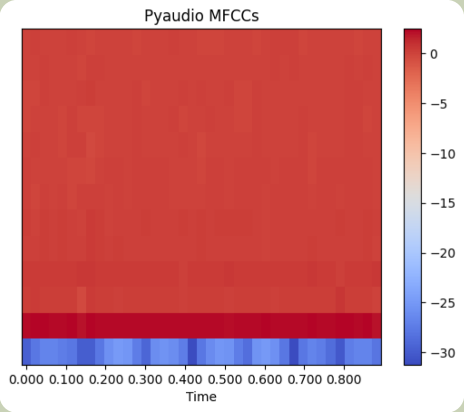
Comparison of libraries

PyAudioAnalysis

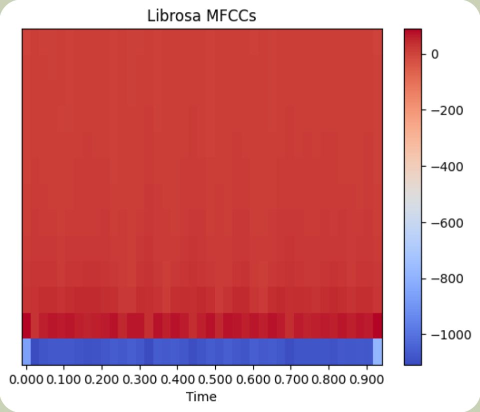
+ deltas

+ chromas

documentation



Librosa



5x faster

+ param MFCCS

documentation

```
librosa.feature.mfcc(*, y=None, sr=22050, S=None, n_mfcc=20, dct_type=2, norm='ortho', lifter=0, **kwargs)
```

PyAudio:

```
def feature_extraction(signal, sampling_rate, window, step, deltas=True):
```

Methodology

04

Modeling

Models used

Table 8. Synthesis of the different models and hyperparameters used for the classification task.

Model name	Parameter	Values
AdaBoost	n_estimators Learning Rate	50, 100 0.1, 1
Extra Trees	n_estimators Max Depth Min Samples Split	100, 200 None, 20 2, 5
Gaussian NB	Var Smoothing	1e-09, 1e-08, 1e-10
Gradient Boosting	n_estimators Learning Rate Max Depth	100, 200 0.1, 0.5 3, 5
K-Neighbors	n_neighbors Weights	5, 10, 15 uniform, distance
LGBM	n_estimators Learning Rate Num Leaves	100, 200 0.1, 0.05 31, 64
Random Forest	n_estimators Max Depth Min Samples Split	100, 200, 300 None, 10, 20 2, 5
XGB	n_estimators Learning Rate Max Depth	100, 200 0.1, 0.05 3, 6

03

Results

Results

RQ1

Eurythmy/ Control

Feature Analysis

Kruskal-Wallis

Features

Slope Sign Changes Ratio

Mean

Variance

PyaudioAnalysis

...

Correlation Treshold: 0.8

Table 9. Kruskal-Wallis test of the feature differences between Control and Eurythmy Groups.

Feature	Control Avg	Eurythmy Avg	p-Value
slope_sign_changes_ratio	0.0297	0.0087	0.00e+00
mfcc_9_mean	0.0534	0.0431	1.81e-239
mfcc_12_mean	0.0426	0.0344	5.81e-225
mfcc_13_mean	0.0303	0.0241	2.29e-207
mfcc_11_mean	0.0400	0.0322	6.01e-206
delta_spectral_spread_std	0.0632	0.0766	8.10e-135
dfa	1.6961	1.6382	1.89e-123
delta_energy_std	0.0509	0.0433	1.66e-35
variance	0.6927	0.4800	8.63e-33
interquartile_range	0.8215	0.6964	6.49e-27
energy_mean	0.1320	0.1526	2.93e-25
cepstra_8_std	0.7639	0.7646	2.38e-23
delta_chroma_8_std	4.51e-05	4.13e-05	6.85e-17
delta_mfcc_12_std	0.1189	0.1205	5.45e-09
mean	-0.0001	-0.0366	1.95e-08
delta_zcr_mean	-3.15e-07	7.33e-08	2.22e-06
delta_chroma_4_mean	2.49e-08	2.47e-08	3.38e-06
chroma_std_mean	0.0551	0.0550	8.27e-06
delta_energy_entropy_mean	2.60e-05	-3.41e-05	1.01e-05
delta_chroma_3_mean	1.04e-06	1.19e-06	3.20e-05
delta_spectral_rolloff_mean	-2.56e-08	1.32e-07	0.0003
cepstra_1_std	1.4264	1.4421	0.0017
delta_spectral_centroid_std	0.0379	-0.0376	0.0023
delta_chroma_std_mean	1.01e-06	-6.95e-07	0.0033
energy_std	0.0833	0.0819	0.0083

Cannonical Comparison

Average

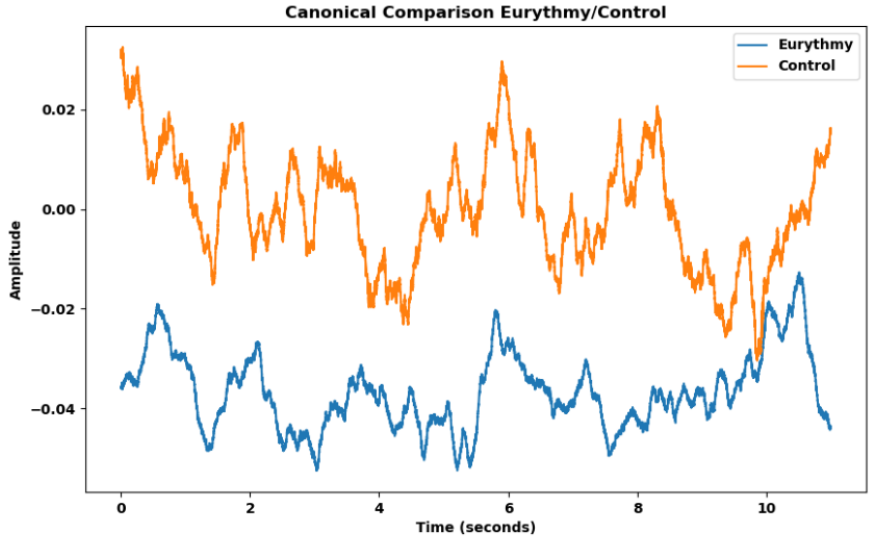


Figure 7. Eurythmy and Control average electrical signals when eurythmy gestures were performed.

Model Results

Table 10. Performance metrics for models in classifying eurythmy and control samples.

Model	F1	Accuracy	Precision	Recall
baseline	-	0.5038	-	-
adaboost	0.7134	0.7153	0.7211	0.7153
extratrees	0.7251	0.7257	0.7277	0.7257
gaussiannb	0.4531	0.5260	0.5558	0.5260
gradientboosting	0.7389	0.7396	0.7422	0.7396
kneighbors	0.5179	0.5179	0.5179	0.5179
lgbm	**0.7488**	**0.7494**	**0.7521**	**0.7494**
randomforest	0.7337	0.7344	0.7367	0.7344
xgb	0.7381	0.7390	0.7422	0.7390

Best Model

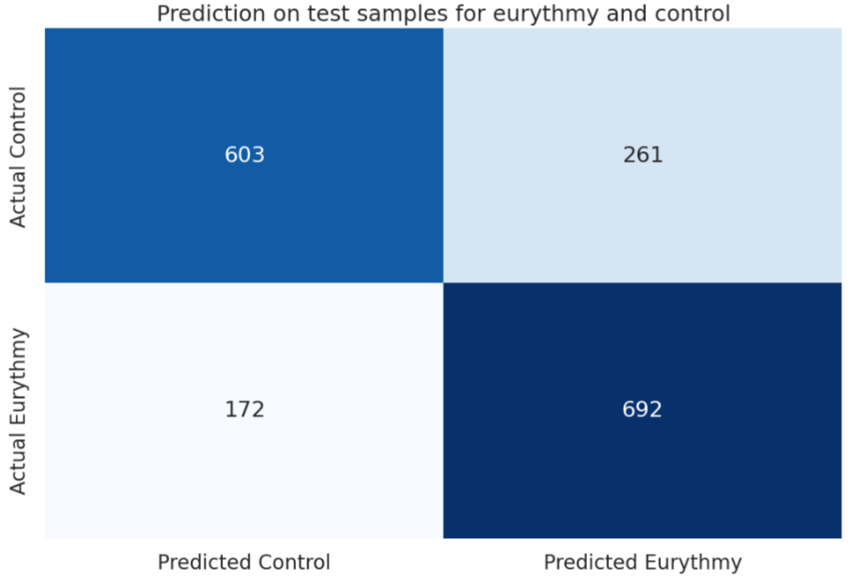


Figure 8. Confusion matrix for the lightGBM (lgbm) model between eurythmy and control.

lightGBM

Test Accuracy: 0.75

+49% baseline

F1-Score: 0.75

Results

RQ2

A/G/D

Letters

A

G

D

Feature Analysis

Kruskal-Wallis

Features

Cepstrals

Mean

Energy

Hjorth

...

Correlation Treshold: 0.9

Table 11. Statistical significances of Kruskal-Wallis between letter groups.

Feature	A Avg	G Avg	D Avg	p-Value
mean	0.0757	-0.0056	-0.0448	2.59e-30
root_mean_square_energy	0.7505	0.7186	0.6903	1.40e-09
cepstra_4_avg	0.2641	0.2623	0.2582	2.23e-09
cepstra_1_std	1.3481	1.3422	1.3313	9.00e-09
zero_crossing_rate	4.70e-05	4.71e-05	4.89e-05	0.0011
dfa	1.5778	1.5745	1.5777	0.0040
delta_chroma_3_std	0.0140	0.0142	0.0137	0.0069
spectral_spread_std	0.0356	0.0362	0.0357	0.0073
slope_sign_changes_ratio	0.0096	0.0102	0.0096	0.0082
spectral_flux_std	0.0719	0.0727	0.0710	0.0113
spectral_centroid_std	0.0235	0.0238	0.0235	0.0116
mfcc_3_mean	0.2130	0.2112	0.2132	0.0118
delta_mfcc_5_std	0.0944	0.0953	0.0952	0.0123
delta_spectral_spread_std	0.0621	0.0632	0.0624	0.0129
cepstra_2_std	1.2621	1.2697	1.2412	0.0198
mfcc_5_mean	0.1373	0.1362	0.1374	0.0231
mfcc_8_mean	0.0784	0.0782	0.0791	0.0317
mfcc_7_mean	0.0769	0.0765	0.0776	0.0340
delta_spectral_centroid_std	0.0389	0.0394	0.0390	0.0387
mfcc_6_mean	0.1447	0.1438	0.1450	0.0500

Cannonical Comparison

Interpolation Resampling and Average

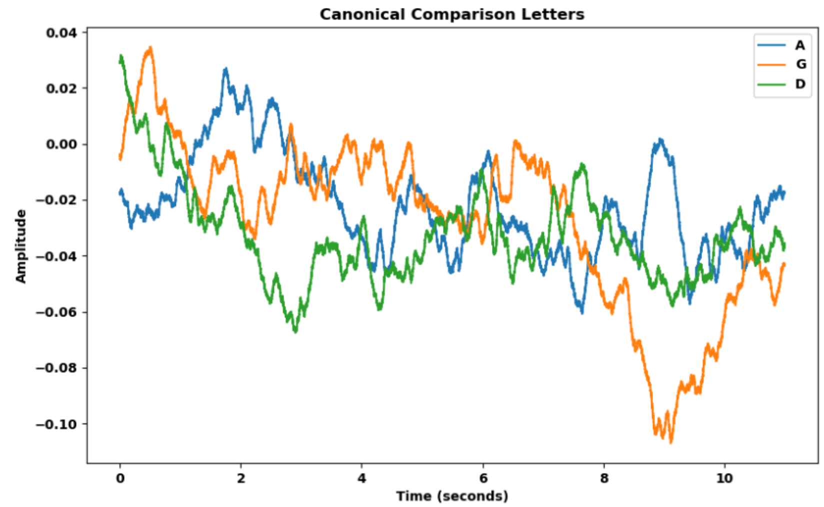


Figure 9. Canonical representation of the average potential curve of each group (A,G,D) after resampling and averaging.

Model Results

Table 12. Performance metrics for models in classifying eurythmy gestures.

Model	F1	Accuracy	Precision	Recall
baseline	-	0.3467	-	-
adaboost	0.4388	0.4392	0.4415	0.4392
extratrees	0.4349	0.4392	0.4363	0.4392
gaussiannb	0.4186	0.4185	0.4192	0.4185
gradientboosting	**0.4536**	**0.4586**	**0.4539**	**0.4586**
kneighbors	0.3738	0.3757	0.3750	0.3757
lgbm	0.4320	0.4350	0.4307	0.4350
randomforest	0.4444	0.4475	0.4435	0.4475
xgb	0.4534	0.4558	0.4530	0.4558

Best Model

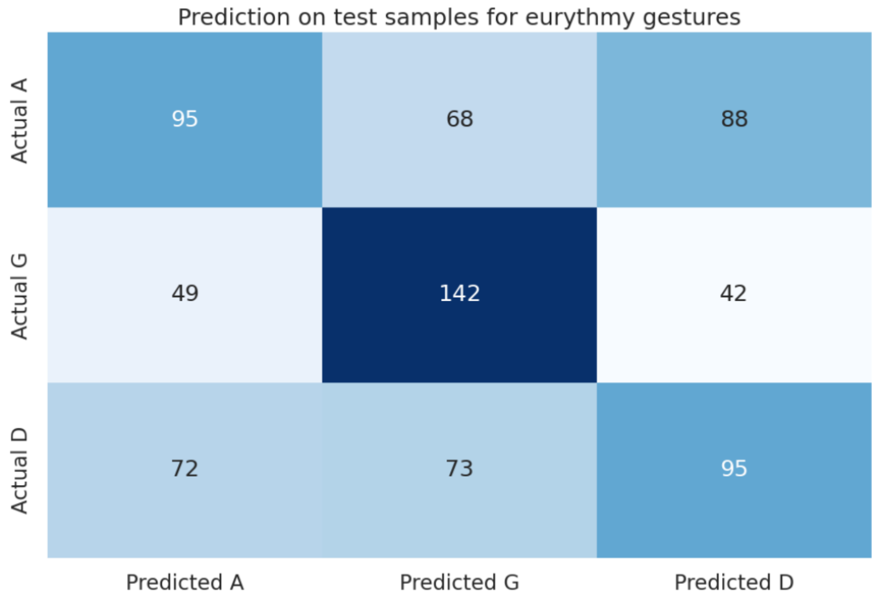


Figure 10. Confusion matrix for the XGBoost (xgb) model between eurythmy letters (A-G-D).

Gradient Boosting

Test Accuracy: 0.46

+32% baseline

F1-Score: 0.45

Results

RQ3

salad/tomato/basil

Feature Analysis

Kruskal-Wallis

Features

Dfa

Slope Sign Changes Ratio

MFCC

Variance

...

Correlation Treshold: 0.8

Table 13. Statistical Analysis of Feature Differences across Plant Species.

Feature	Lettuce Avg	Tomato Avg	Basil Avg	p-Value
dfa	1.6519	1.6271	1.6783	0.00e+00
slope_sign_changes_ratio	0.0220	0.0152	0.0260	3.54e-198
delta_mfcc_3_std	0.1085	0.1136	0.1062	1.15e-118
mfcc_6_std	0.0531	0.0554	0.0526	1.11e-105
mfcc_5_std	0.0510	0.0531	0.0505	1.92e-95
zcr_mean	0.0004	0.0004	0.0007	4.92e-91
mfcc_10_std	0.0472	0.0494	0.0470	3.37e-83
energy_entropy_std	0.1260	0.1342	0.1303	9.85e-83
delta_mfcc_8_std	0.0837	0.0870	0.0827	7.63e-80
delta_mfcc_7_std	0.0795	0.0828	0.0790	5.93e-75
variance	0.1940	0.1602	0.1780	1.48e-74
delta_mfcc_12_std	0.0729	0.0762	0.0726	2.25e-74
mfcc_9_std	0.0463	0.0484	0.0462	2.65e-74
mfcc_13_std	0.0385	0.0404	0.0389	4.10e-72
delta_mfcc_11_std	0.0770	0.0804	0.0764	9.50e-72
delta_energy_std	0.0788	0.0833	0.0820	2.63e-65
delta_chroma_11_std	0.0025	0.0027	0.0025	1.76e-61
delta_spectral_rolloff_std	0.0040	0.0043	0.0042	1.23e-57
delta_chroma_2_std	7.66e-05	8.14e-05	7.80e-05	7.79e-56
delta_chroma_3_std	0.0131	0.0139	0.0137	5.93e-55
delta_spectral_flux_std	0.1126	0.1178	0.1108	3.45e-51
cepstra_1_std	1.3283	1.3552	1.3269	2.29e-50
interquartile_range	0.3874	0.3811	0.4151	1.00e-49
energy_mean	0.2299	0.2210	0.2145	1.10e-43
cepstra_4_avg	0.2565	0.2648	0.2555	7.12e-38
spectral_spread_std	0.3321	0.3032	0.3156	1.14e-37
root_mean_square_energy	0.7415	0.7562	0.7344	4.99e-37
energy_std	0.0612	0.0632	0.0623	2.83e-32
hjorth_mobility	0.0001	0.0001	0.0001	4.94e-32
delta_spectral_flux_mean	0.0048	0.0051	0.0050	3.33e-26
hjorth_complexity	11097.4615	11059.9986	11611.5156	5.48e-17
chroma_std_mean	0.0550	0.0550	0.0552	2.80e-14
mean	0.0076	0.0029	0.0126	7.16e-06
mfcc_10_mean	0.0568	0.0574	0.0565	6.16e-05
mfcc_11_mean	0.0382	0.0388	0.0385	0.0022
mfcc_3_mean	0.2311	0.2326	0.2317	0.0169
mfcc_9_mean	0.0514	0.0519	0.0514	0.0204
delta_mfcc_5_mean	3.17e-05	-7.96e-05	1.78e-05	0.0241
delta_mfcc_8_mean	4.80e-05	-3.96e-05	-1.37e-05	0.0394

Cannonical Comparison

Average

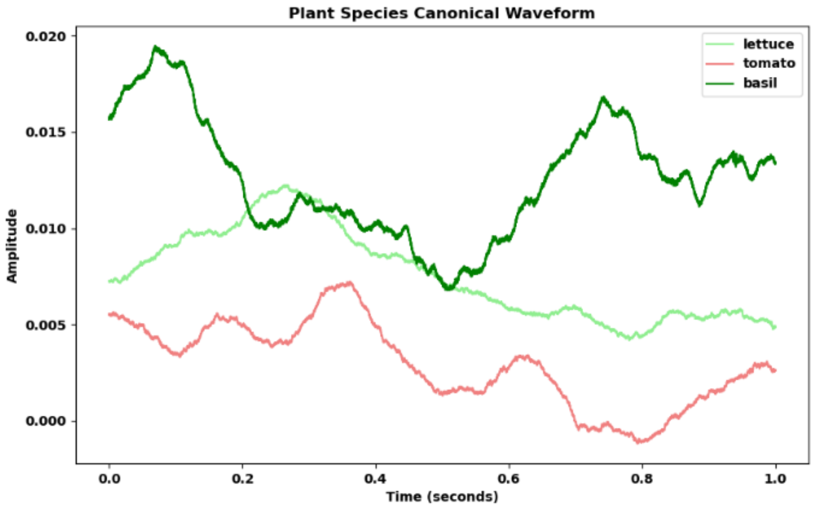


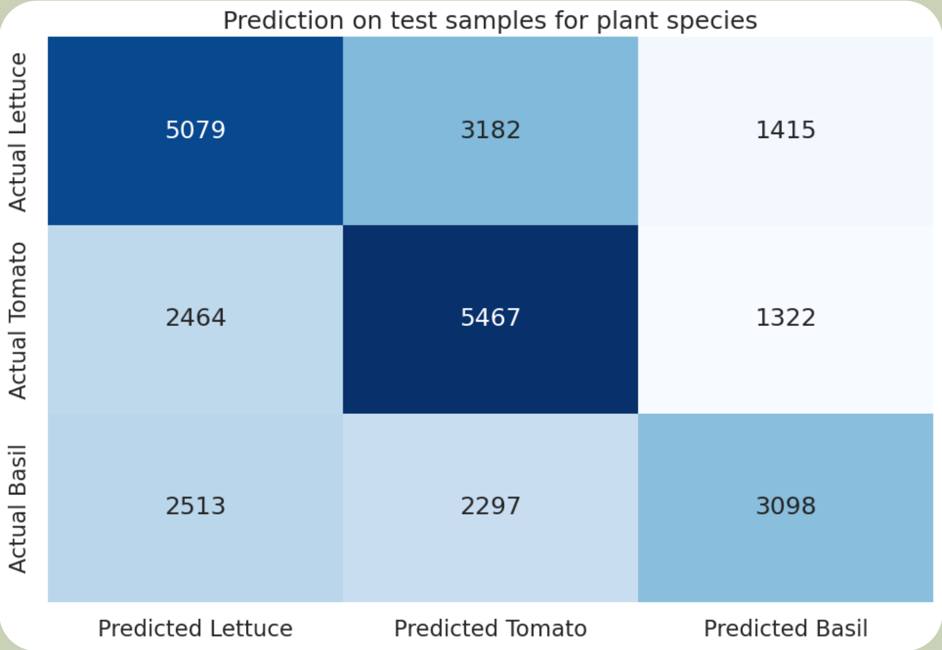
Figure 11. Canonical representation of voltage curves of lettuce, tomato, and basil .

Model Results

Table 14. Model Evaluation Metrics.

Model	F1	Accuracy	Precision	Recall
baseline	-	0.3605	-	-
adaboost	0.4136	0.4180	0.4186	0.4180
extratrees	0.4225	0.4299	0.4355	0.4299
gaussiannb	0.2759	0.3670	0.3870	0.3670
gradientboosting	0.4949	0.4981	0.5005	0.4981
kneighbors	0.3382	0.3515	0.3443	0.3515
lgbm	0.4849	0.4868	0.4881	0.4868
randomforest	0.4474	0.4534	0.4601	0.4534
xgb	**0.5051**	**0.5084**	**0.5108**	**0.5084**

Best Model



XGBoost

Test Accuracy: 0.5

+41% baseline

F1-Score: 0.5

04

Discussion

Discussion

01

Featurization

Processing Possibilities

Downsampling

What rate should we choose?

Filter

Should we remove high frequencies?

Offset

When should we center the wave?

Processing

What should we input in the model?

Iterations: Sample Rate | Filter Threshold | Offset Moment | Input

MFCC parameters

01 Framing
- Window Size: 20ms - 25 ms (audio)
- Hop Length: 0.50ws – 0.75ws

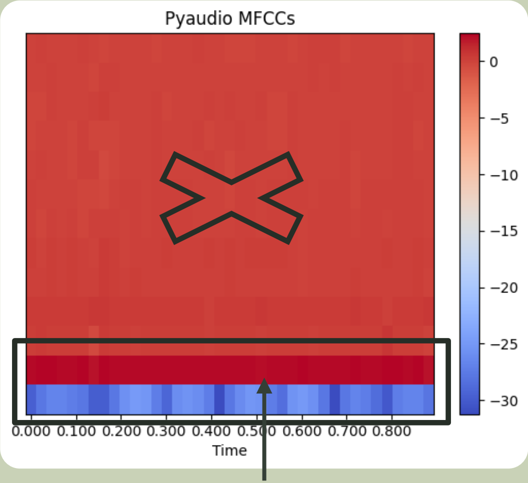
02 Windowing
- Window Function: Hamming | Hann

03 FFT
- Nº of Points FFT: 512 | 1024 | 2048

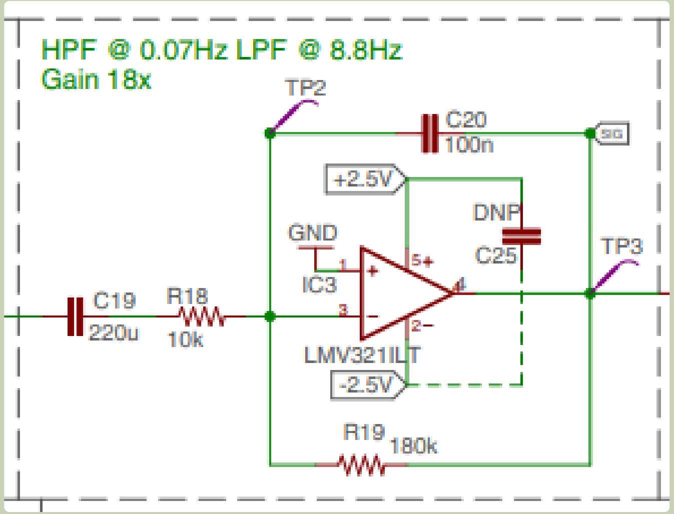
04 Mel Filterbank
- Nº of Filterbanks: 20 - 40

05 MFCCs
- Nº of MFCCs: Type-II | None

MFCC idea



Look closer here



Discussion

02

Modeling

Input Approaches

01

MFCCs

- Machine Learning (+ features)
- MFCC- CNN (+ResNet)
- FCNN

02

Spectrogram

- MFCC- CNN (+ResNet)

03

Signal

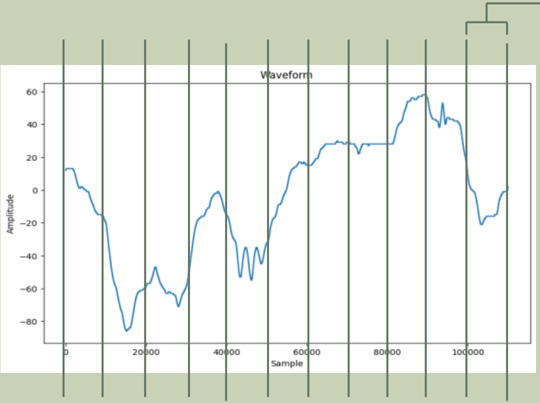
- LSTM Classifier

x2 time series

This was Tried

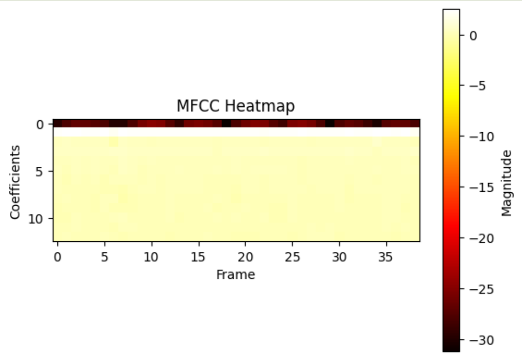
Neural Networks (Didn’t Work)

Similar Data Input



Trim wave in **1s** segments

Output Matrix



Size: 13x39 (507 elem)

Data points

6.7k

↓

75k

FCNN & MFCC-CNN

Discussion

03

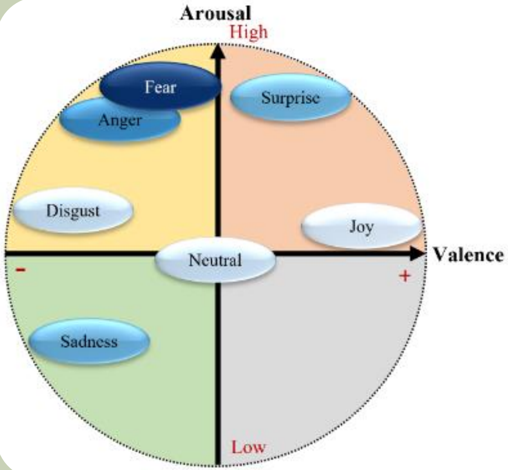
Future of Work

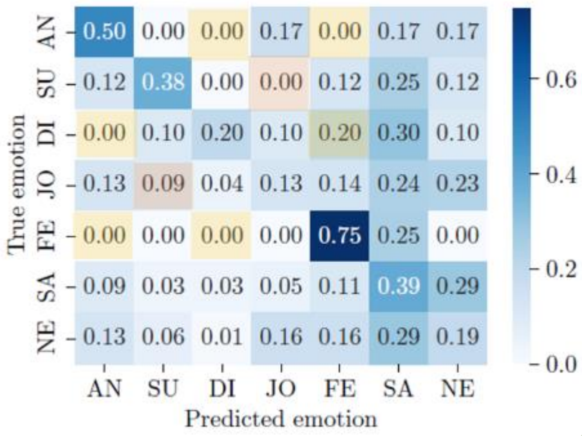
Human emotion

Kruse et al, 2023

Species	Output	Accuracy
Basil	7 emotions	0.32

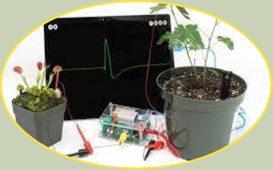
Model	Test set Accuracy	Test set Recall
MLP	0.399	0.220
biLSTM	0.260	0.351
MFCC-CNN	0.377	0.275
MFCC-RestNet	0.318	**0.324**





Vision

Bio-Lingo



Elings, 2016

Effects of plants on People

physical

1. Stress and mental fatigue relief
2. Treatment of mental health issues
3. Lower blood pressure and heart rate
4. Faser recovery from stress
5. Reduction in risk factors for diseases

mental

6. Improved self-steem and responsibility
7. Enhanced tranquility and enjoyment
8. Promotion of relaxation and reflection
9. Positive correlation with well-being

social

10. Stimulation of social cohesión
11. Enhanced social interaction
12. Reduction in lonileness

Dissemination

Deciphering Human Motion

Presentation

Date

Dec 12, 6pm-7pm

Location

26 Trowbridge St

RCC.HARVARD.EDU

RCC

https://rcc.harvard.edu/event/made-upm

GitHub Project

https://github.com/alvaro-francisco-gil/Plant-Reactivity-Analysis

COOKIECUTTER

Reproducible

Explainable

Scalable

Can plants percieve human gestures?

Article

MDPI

Article

Can plants perceive human gestures? Using AI to track eurythmic human-plant interaction

Alvaro Francisco Gil [1,2], Moritz Weinbeer [3] and Peter A. Gloor [1,*]

[1] MIT Center for Collective Intelligence, Cambridge MA, USA
[2] UPM Technical University of Madrid, Spain
[3] Foundation Fintan, Rheinau, Switzerland
* Correspondence: pgloor@mit.edu

Abstract: This paper explores if plants are capable of responding to human movement by changes in their electrical signals. Towards that goal we conducted a series of experiments, where humans over a period of 6 months were performing different types of Eurythmic gestures in proximity of garden plants, namely salad, basil, and tomatoes. To measure plant perception, we used the plant spikerbox, a device that measures changes in voltage differentials of plants between roots and leaves. Using machine learning, we find that the voltage differentials over time of the plant predict if (a) Eurythmy has been performed, and (b) which kind of Eurythmy gestures has been performed. We also find that the signals are different based on the species of the plant. In other words, the perception of a salad, tomato, or basil might differ just as perception of different species of animals differ. This opens new ways of studying plant ecosystems, while also paving the way to use plants as biosensors for analyzing human movement.

Keywords: plant-human interaction; signal processing; plant action potentials; machine learning; eurythmy; plant bio-sensors

Thanks!

Do you have any questions?

alvaro.francisco.gil@alumnos.upm.es
alvaro_f@mit.edu

POLITÉCNICA