CLASSIFICATION OF MUSICAL INSTRUMENTS

Under the guidance of
Prof. Preeti Rao
Department of electrical Engineering
Indian Institute of Technology, Bombay

Presented by
Nihar Mahesh Gupte
213070002
Harsh Diwakar
213070018

Overview

- Instrument Classification
- Dataset
- Feature Extraction
- Methodology
- Results
- Discussion
- Conclusion

Instrument Classification

- Tons of digital audio material on the Internet
- Applications
  - Indexing Annotation and transcription of database
  - Knowing various musical styles, playlist generation
  - Video scene analysis
- Method
  - Dataset
  - Lists of features
  - Learning Algorithms
  - Performance parameters

Dataset

- Instrument recognition in musical audio signals (IRMAS) dataset
- Contains 11 Classes
  - Cello (cel), Clarinet (cla), Flute (flu), Acoustic guitar (gac), Electric guitar (gel), Organ (org), Piano (pia), Saxophone (sax), Trumpet (tru), Violin (vio), and Human singing
- Training: 6705 audio files, 16 bit stereo at 44.1 kHz
- Testing: 2874 audio samples divided into three parts

Source: A comparison of sound segregation techniques for predominant instrument recognition in musical audio signals, J J Bosch, J Janer, F Fuhrmann, and P Herrera.

Feature Extraction

- Short Time Zero Crossing Rate (ZCR)
- Short Time Energy (STE)
- Mel Frequency Cepstral Coefficients (MFCC)
- Spectral Centroid (SC)
- Spectral Roll off (SRO)
- Spectral Contrast (SCR)

Zero Crossing Rate and Short Time Energy

- ZCR is calculated by counting number of times that time domain signal crosses zero within a short time window

$$Z_n = \sum_{m=-\infty}^{m=\infty} |\operatorname{sgn}(x[m]) - \operatorname{sgn}(x[m-1])|\, w[n-m]$$

Where,

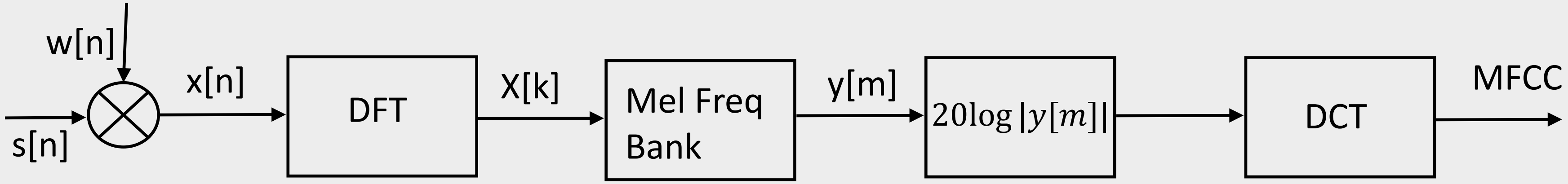
$\operatorname{sgn}(x)$ is sign function and w is window with length L

- STE of speech signal gives information about amplitude variation.

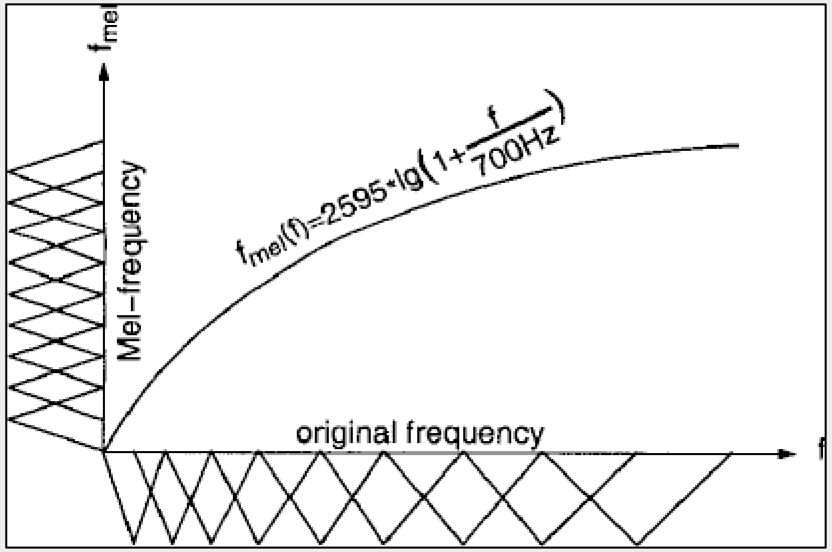
$$E_n = \sum_{m=-\infty}^{m=\infty} (x[m]w[n-m])^2$$

Source: Implementation of ZCR and STE techniques for the detection of the voiced and unvoiced signals in Continuous Punjabi Speech, Anu Priya Sharma

Mel Frequency Cepstral Coefficients



- Frequency to mel transform: $m = 2595\log_{10}(1 + \frac{f}{700})$
- First 13 coefficients are used
- Exploits the property that humans can't distinguish finely at higher frequencies.
- All 13 coefficients are uncorrelated



Source: Computing Mel-frequency cepstral coefficients on the power spectrum, S. Molau, M. Pitz, R. Schluter and H. Ney

Spectral Centroid

- SC indicates the location of center of mass of the spectrum
- It is given as[1]:

$$F_c = \frac{\sum_{k=0}^{k=N-1} f(k)S(n,k)}{\sum_{k=0}^{k=N-1} S(n,k)}$$

Where,

$f(k)$ is the frequency of n^{th} frame

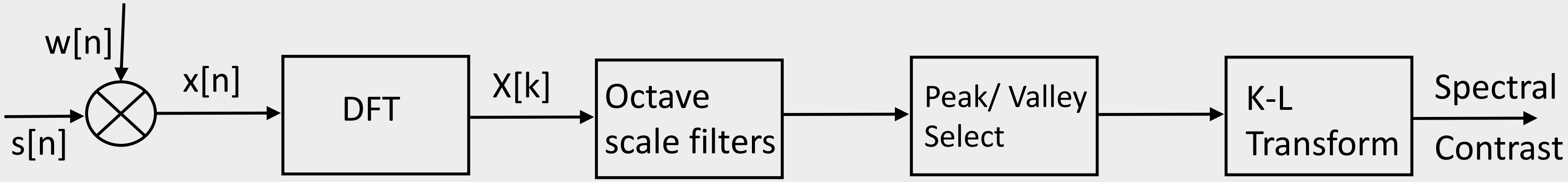
$S(n,k)$ is magnitude spectrum

- Good Predictor of 'brightness' of a sound
- Used as an automatic measure of musical timbre[2]

Source: [1] Signal processing methods for music transcription, Klapuri, A., & Davy, M.
[2] Perceptual effects of spectral modifications on musical timbres, Grey, John M.; Gordon, John W.

Spectral Roll off and Spectral Contrast

- SRO is the frequency below which a specified percentage of the total spectral energy lies[1].
- SCR considers the spectral peak, spectral valley and their difference in each sub-band[2].



$$Peak_k = \log\{\frac{1}{\alpha N}\sum_{i=1}^{\alpha N} x_{k,i}\}, \text{ and } \mathrm{Valley}_k = \log\{\frac{1}{\alpha N}\sum_{i=1}^{\alpha N} x_{k,N-i+1}\}$$

- Then Spectral Contrast is given as: $SC_k = Peak_k - Valley_k$

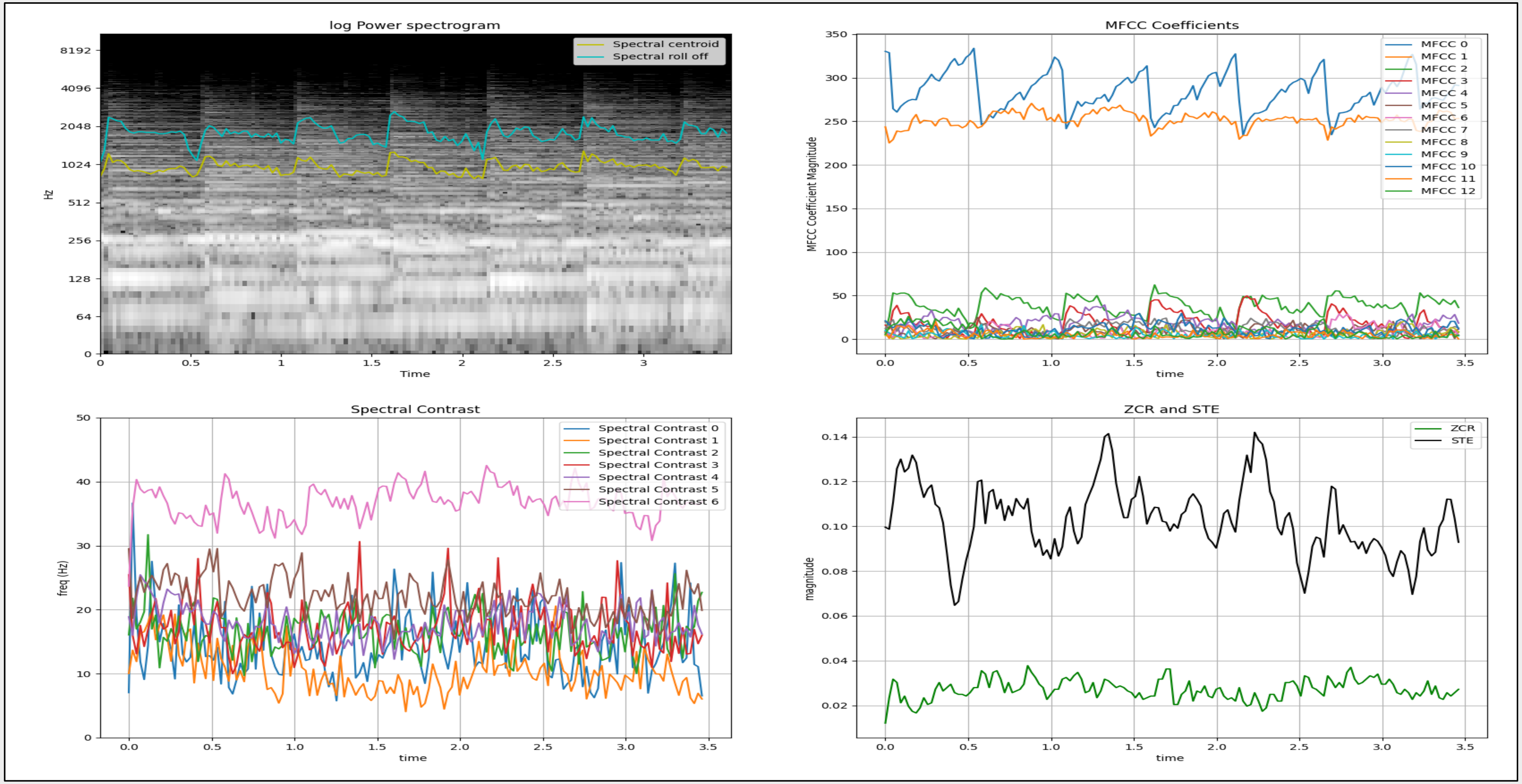
Where,

N is the total number of k^{th} sub-band, $N = 6$ by default in Librosa

α is set to 0.02 in real implementation according to Jiang et.al.

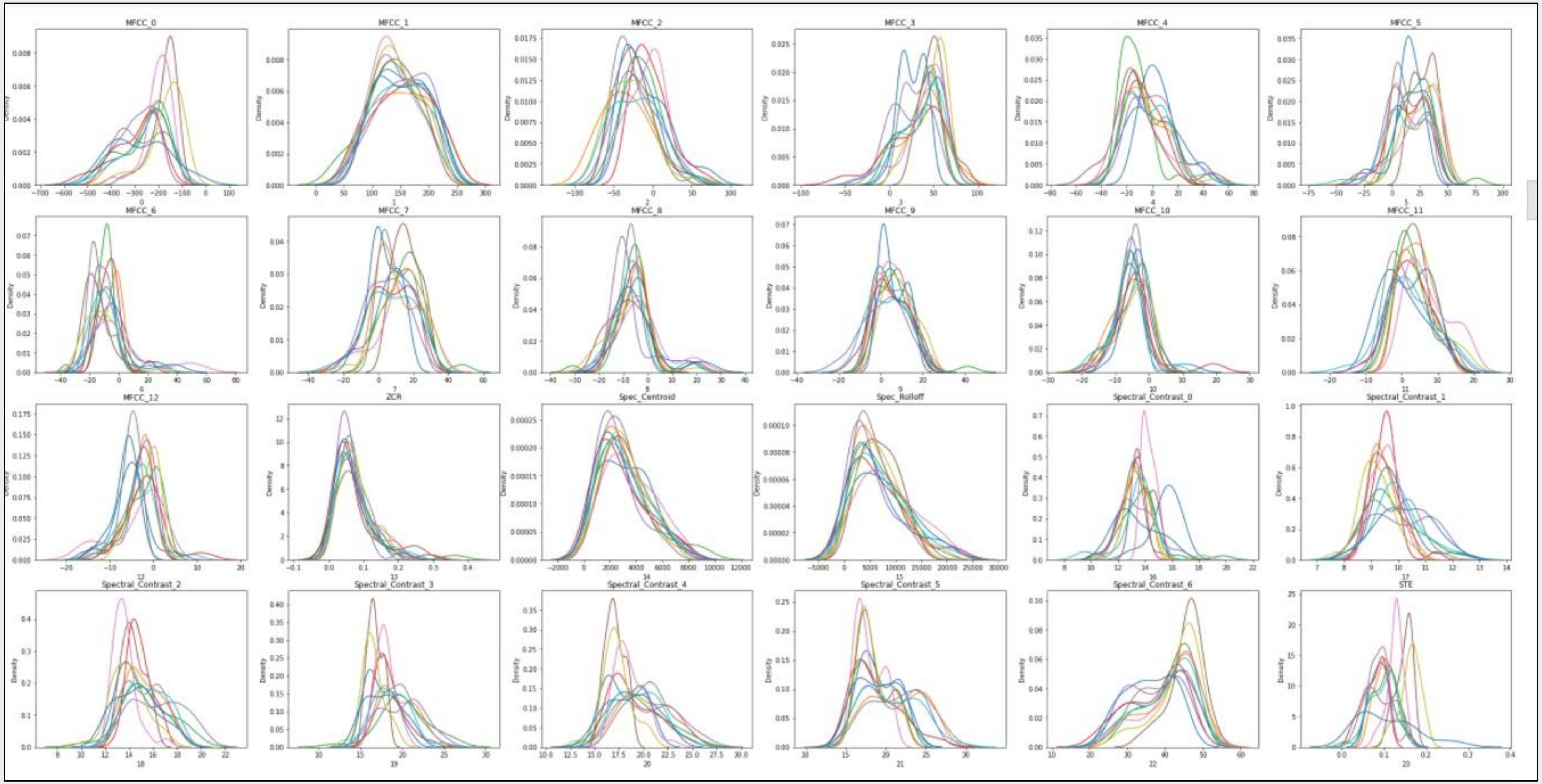
Source: [1] librosa: Audio and music signal analysis in python, McFee, Brian, C. Raffel, D. Liang, D. PW Ellis, M. McVicar, E. Battenberg, and O. Nieto
[2] Music type classification by spectral contrast feature, D.N. Jiang, L. Lu, H.J. Zhaung, J.H. Tao and L.H. Cai

Visualizing the Features



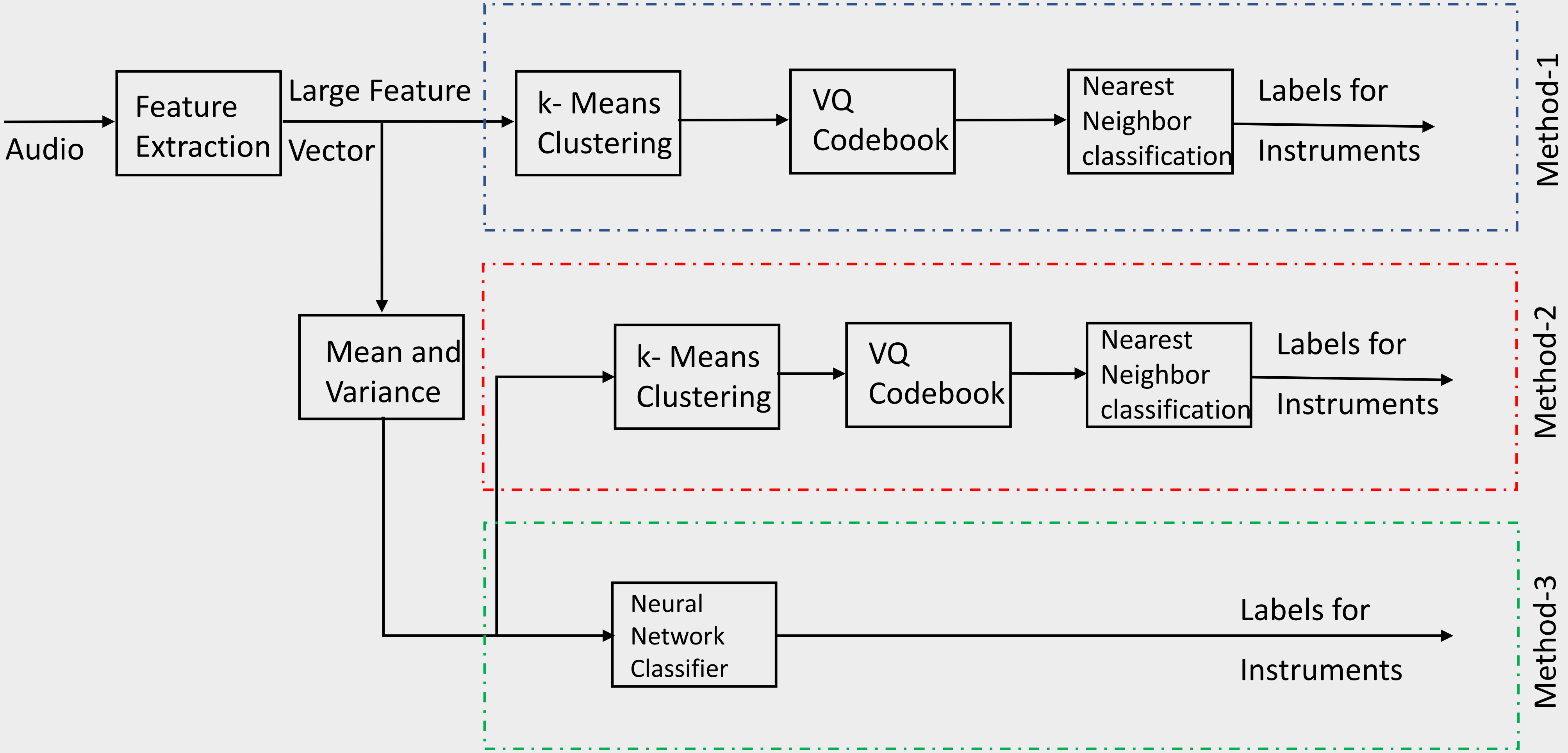
Feature visualization for a sample from class Piano

Visualizing the Features (contd.)



Probability Density Function for all the extracted features

Methodology



Methodology (contd.)

- Method-1
  1. Dataset was converted to data frames, with 20 ms hop and 30 ms window.
  2. Each frame provides one training example.
  3. No need of end pointing as there was little or no silence in the dataset.
  4. Feature extracted using librosa : 13 MFCC, 1 Spectral rolloff, 7 spectral contrast, 1 zero crossing rate, 1 Short time Energy, 1 Spectral centroid.
  5. Followed by K-Means clustering for K = 32,64,128,512,1024.
  6. Finally, test set also decomposed into features and followed by Nearest Neighbors rule to predict classes.
- Method-2
  1. After the 4th step from method-1, Mean and variances of all features were taken for an entire sample and concatenated to make a 24 + 24 = 48 dimensional vector.
  2. Again steps 5-6 were followed from method-1 to classify the instrument class
- Method-3
  1. Dataset of method-2 was fed into a neural network with 2 hidden layers, having 64 and 32 neurons each.

Evaluation Metrics

- Evaluation was performed on a test dataset with 807 samples.
- $Accuracy = \frac{True\ Postives}{Total\ Samples}$
- True positives
  - Test dataset contains audios with multiple labels.
  - We are counting the hypotheses as true positive even when our model gives any of the true labels
  - Also, to compensate for the above effect we are adding all the labels as false negative when our model fails to provide the correct label.
- Confusion matrix
  - Evaluates the model performance using a cross- tabulation of actual and predicted classes

Predicted Labels

	Positive	**Negative**
Positive	True Positives	False Negatives
Negative	False Positive	True Negatives

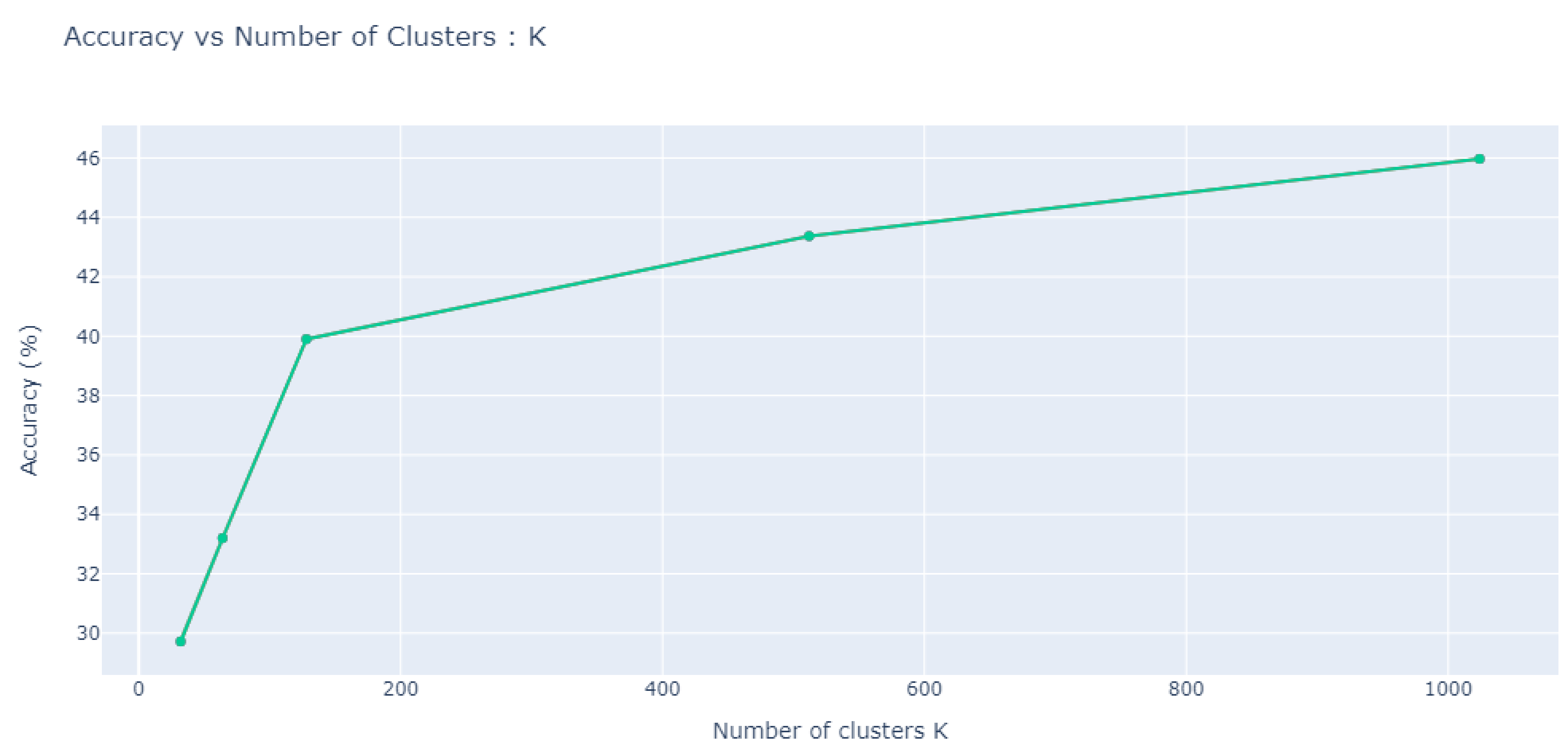
True Labels

Results

Model	Hyperparameter	Accuracy (%)
Extracted Features+ VQ CB Matching (Method-1)	k = 32	29.72%
	k = 64	33.20%
	k = 128	39.90%
	k = 512	43.37%
	k = 1024	45.97%
Mean Variance+ VQ CB Matching (Method-2)	k = 16	22.30%
	k = 32	20.07%
	k = 64	18.46%
	k = 128	19.20%
Mean Variance+ Single layer Neural Network (Method-3)	Neurons in layer-1: 64 Neurons in layer-2: 32 Neurons in layer-3: 11	65.05%

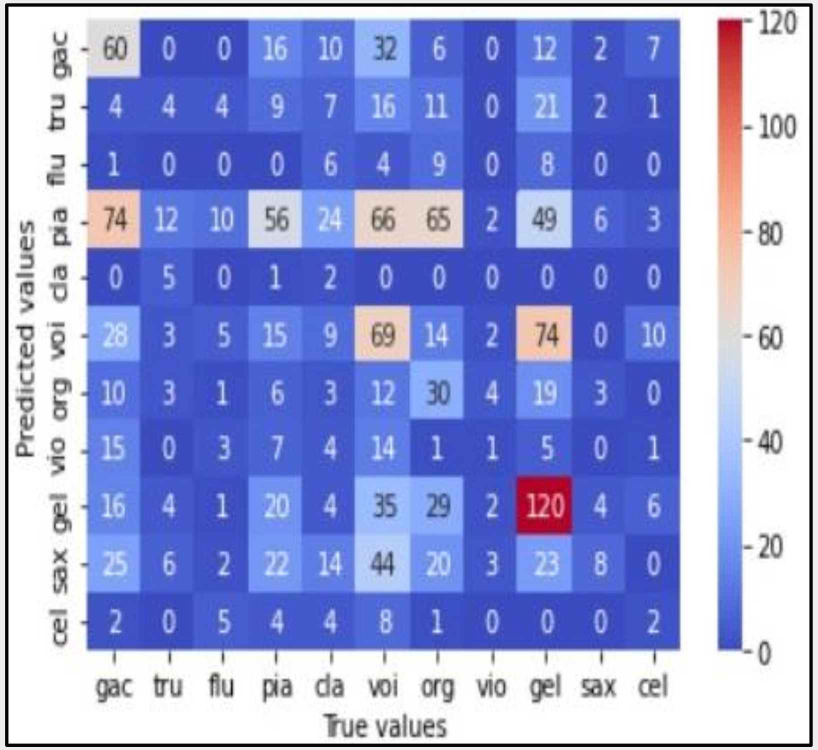
Accuracy for different models and hyperparameters

Results (contd.)

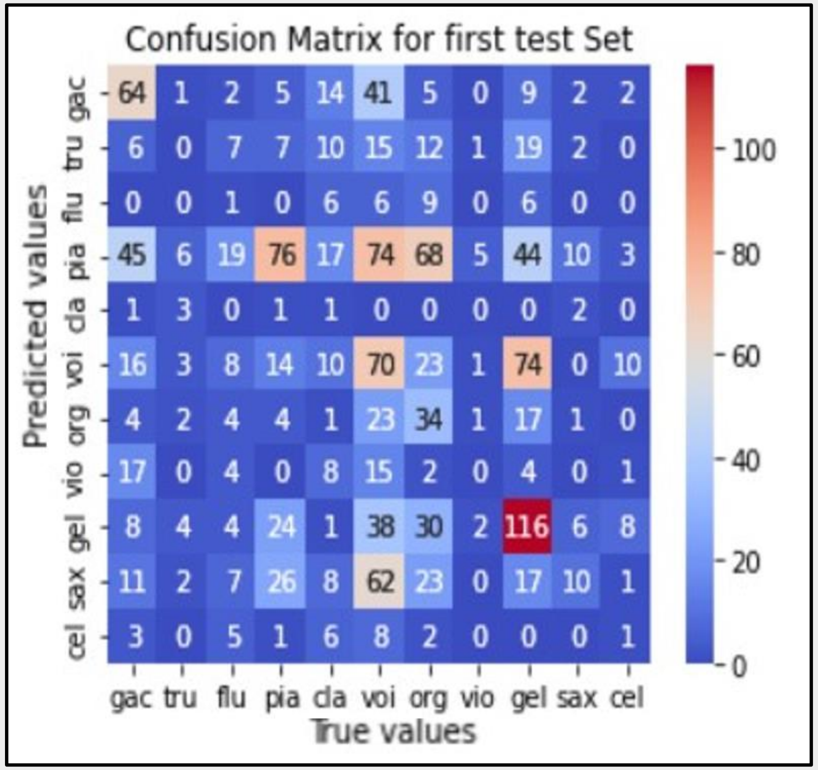


Accuracy vs Number of clusters for Method-1

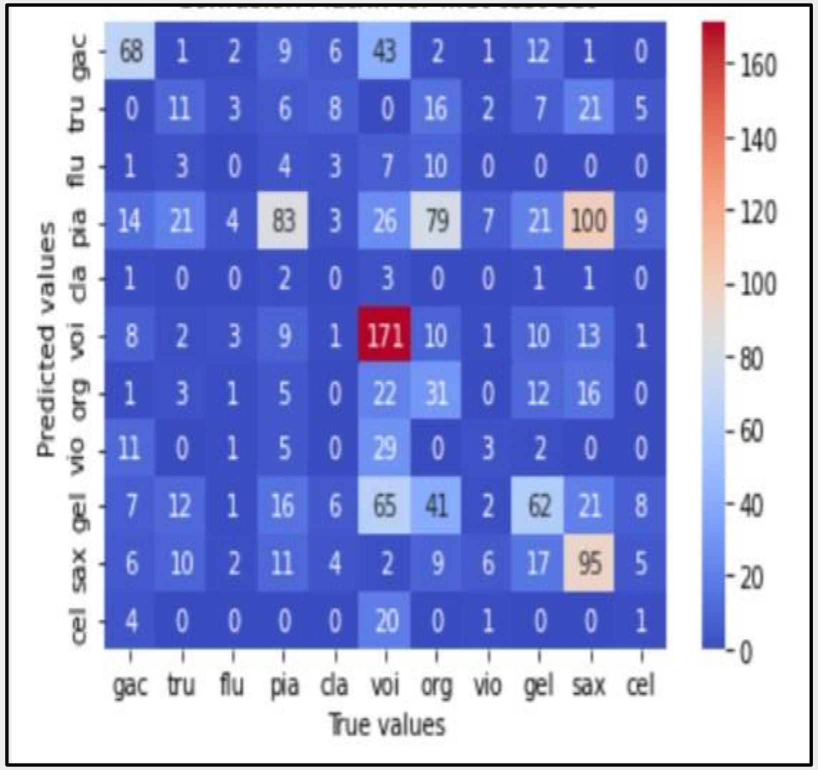
Results (contd.)



Confusion matrix for Method-1, k=512



Confusion matrix for Method-1, k=1024



Confusion matrix for Method-3

Discussion

- Accuracy was increasing as k increases for method-1.
- However, time and space taken for execution was also increasing.
- Using elbow method, the best k(number of templates in VQ Codebook) is 128 as after this value there is very slight improvement in Accuracy.
- Method-2 was poor in terms of accuracy because only the mean and variance of features across frames were used instead of all frames.
- Features used had different mean and variances, which were exploited in method-2.
- Speech features along with neural network not only gave a better result than a very simple template matching technique but also the computational time was less.

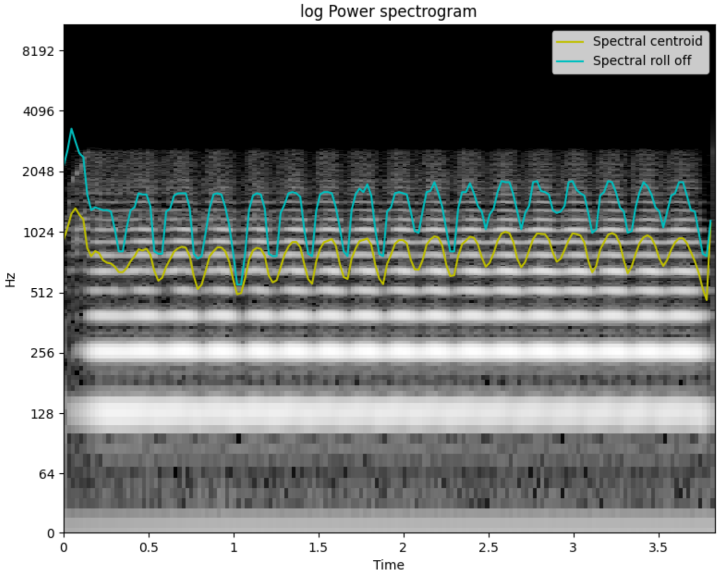
Challenges and Future Works

- Challenges:
  - The dataset to be used initially was TensorFlow Nsynth which was a huge dataset and the instrument sounds were pure.
  - However, we lacked the computational power to handle such a huge dataset.
  - Continuously changing features and hyperparameters (k) to get the best predictions because of slow computation as k kept increasing.
- Future Works:
  - Implement the classification task with the features extracted and more complex classifiers.
  - Also using the TensorFlow Nsynth dataset for building an even better classifier.

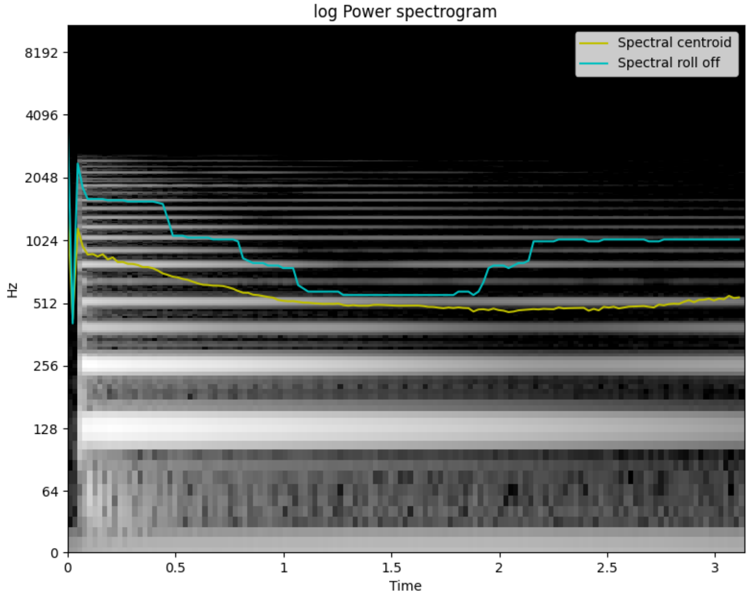
Conclusion

- For classification of musical instruments, several temporal, spectral and coefficient space features were explored and extracted from audio file.
- After retrieving the features, these features were used to classify the instruments using template matching and statistical learning methods.
- For template matching methods, accuracy was increasing as the number of vectors in VQ Codebook was increasing but so was the computation time.
- Even for a very less set of features (mean and variance of all the features across all frames), neural network classifiers performed better than template matching techniques
- Thus, feature extraction along with modern machine learning can provide excellent results in future.

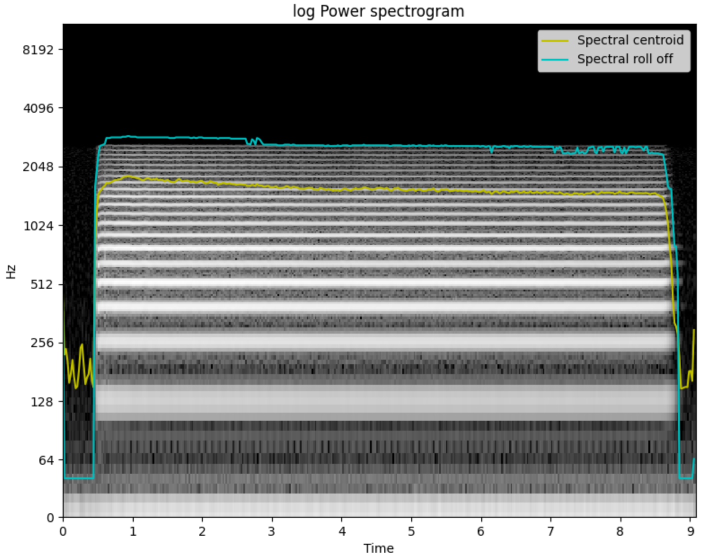
Thankyou!



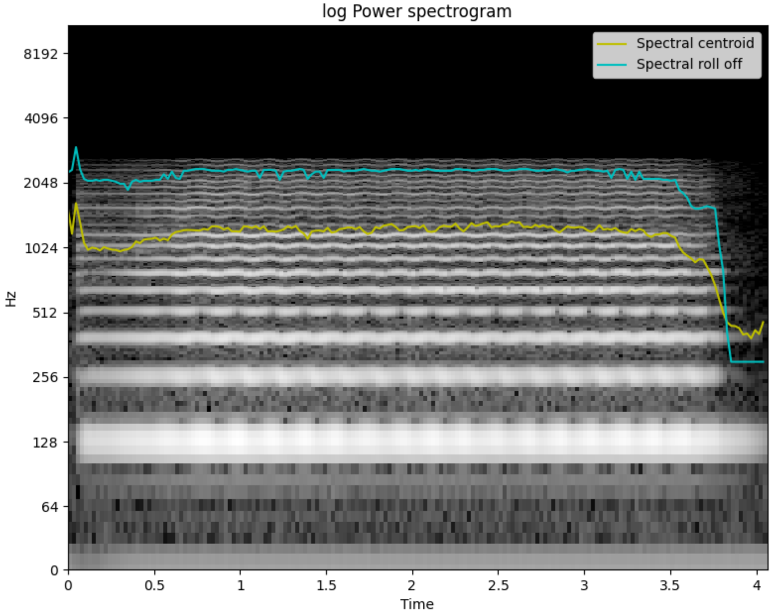
Flute



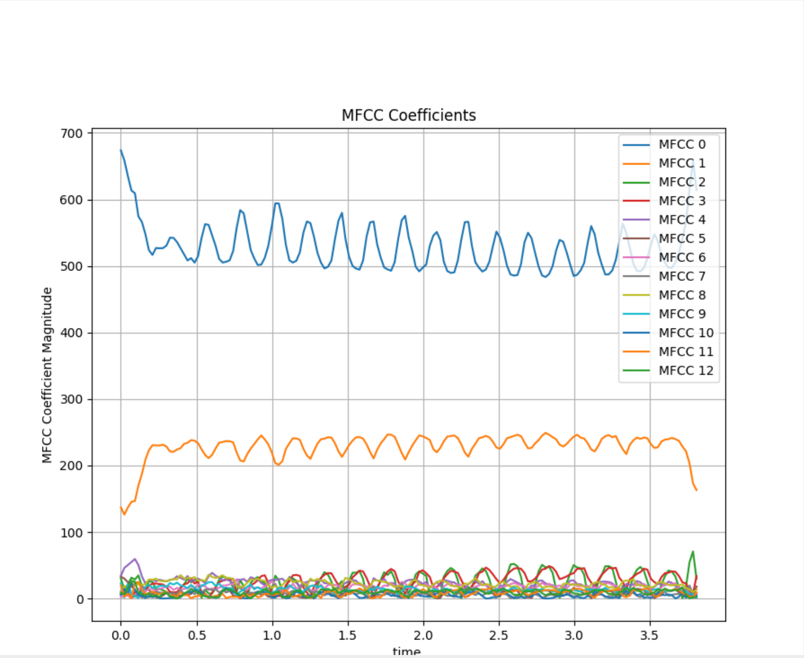
Piano

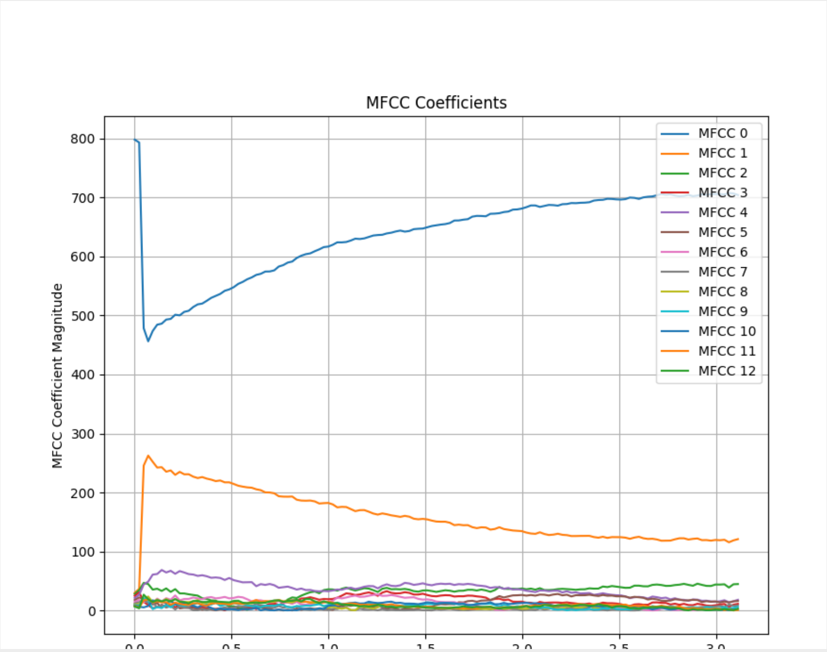


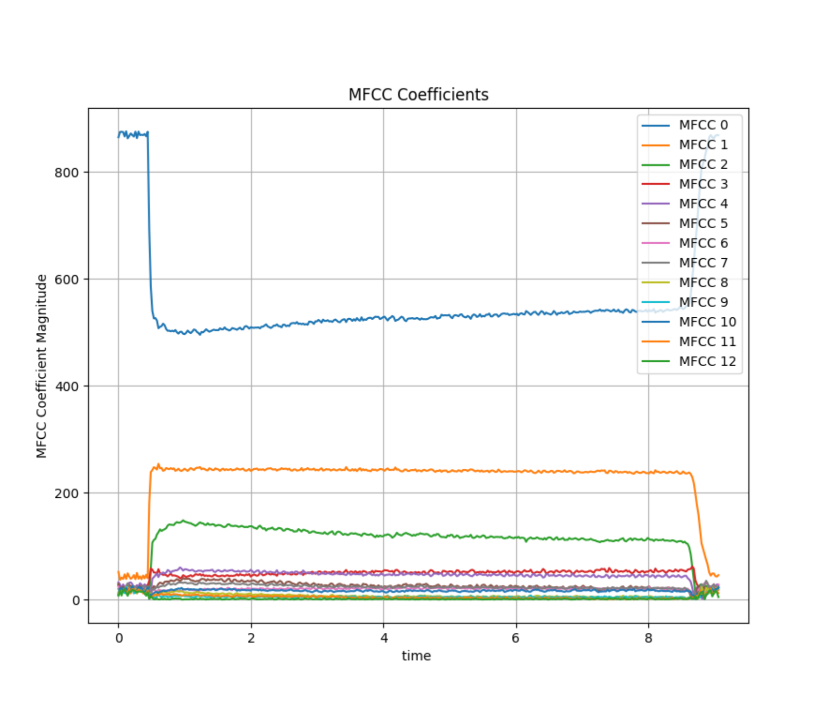
Trumpet

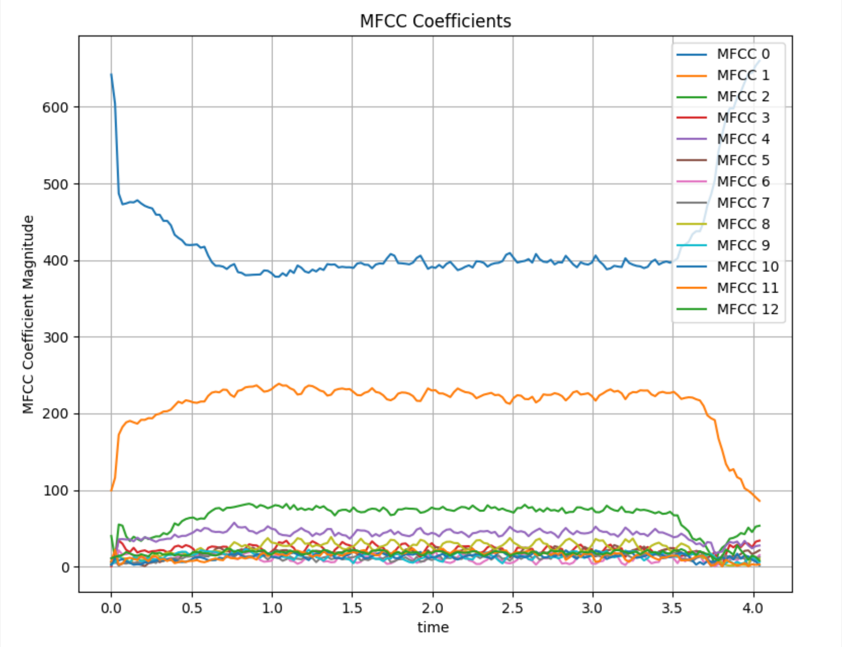


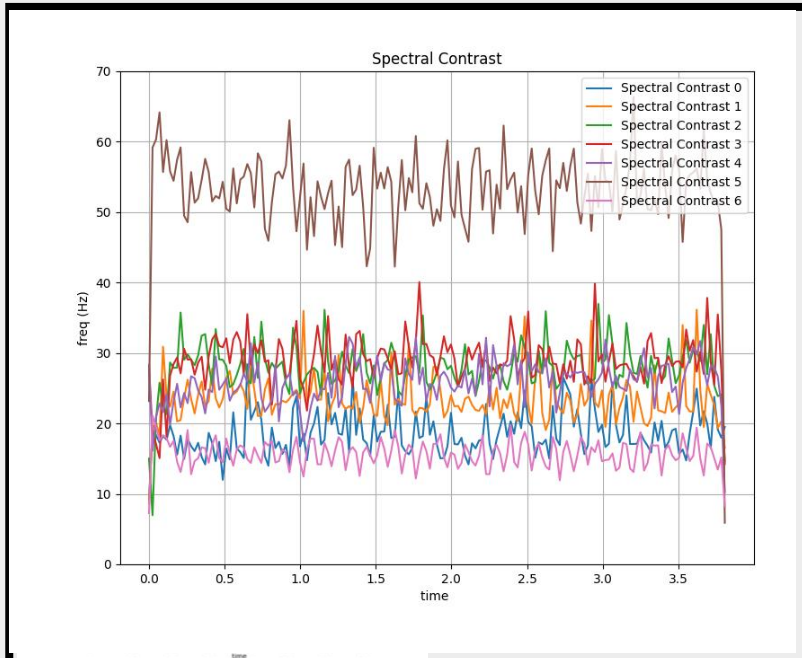
Violin

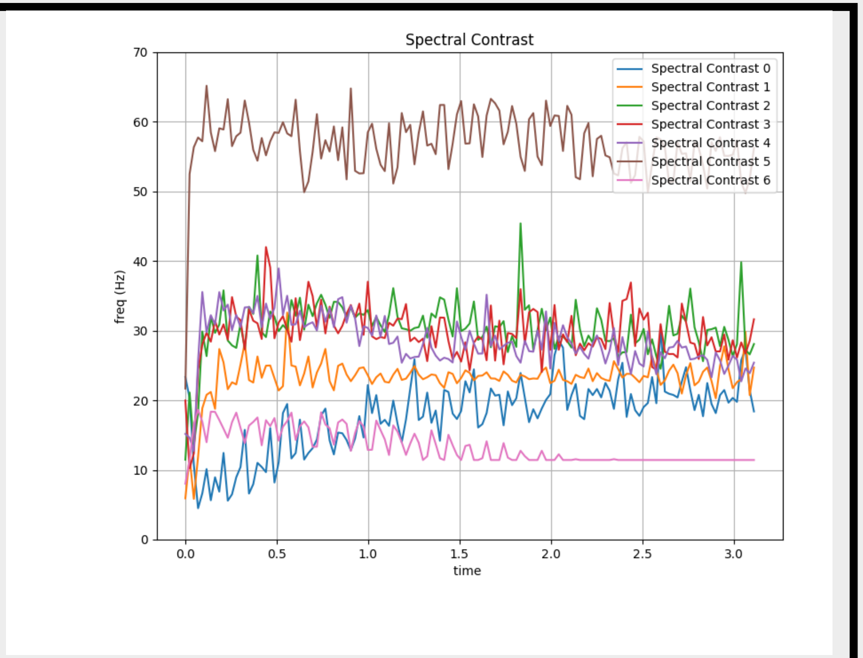

MFCC Coefficients
MFCC Coefficient Magnitude
time
MFCC 0
MFCC 1
MFCC 2
MFCC 3
MFCC 4
MFCC 5
MFCC 6
MFCC 7
MFCC 8
MFCC 9
MFCC 10
MFCC 11
MFCC 12

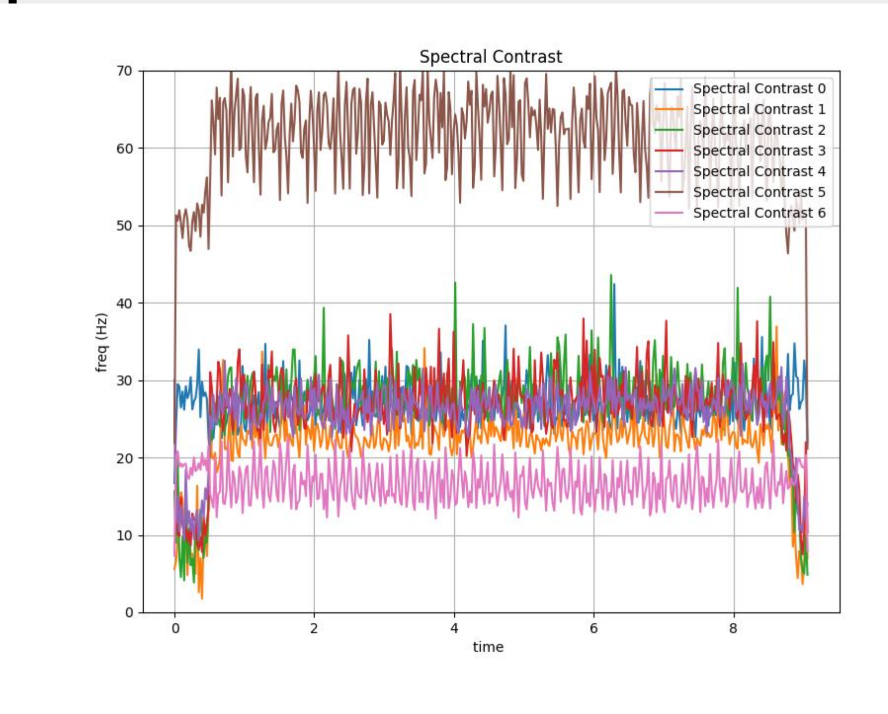

MFCC Coefficients
MFCC Coefficient Magnitude
MFCC 0
MFCC 1
MFCC 2
MFCC 3
MFCC 4
MFCC 5
MFCC 6
MFCC 7
MFCC 8
MFCC 9
MFCC 10
MFCC 11
MFCC 12

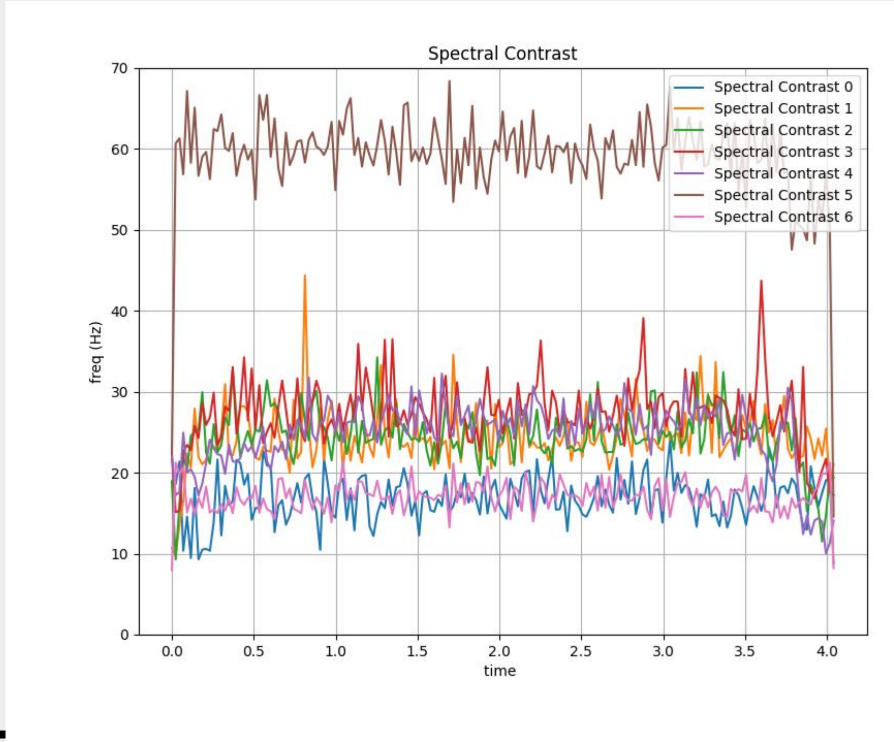

MFCC Coefficients
MFCC Coefficient Magnitude
time
MFCC 0
MFCC 1
MFCC 2
MFCC 3
MFCC 4
MFCC 5
MFCC 6
MFCC 7
MFCC 8
MFCC 9
MFCC 10
MFCC 11
MFCC 12

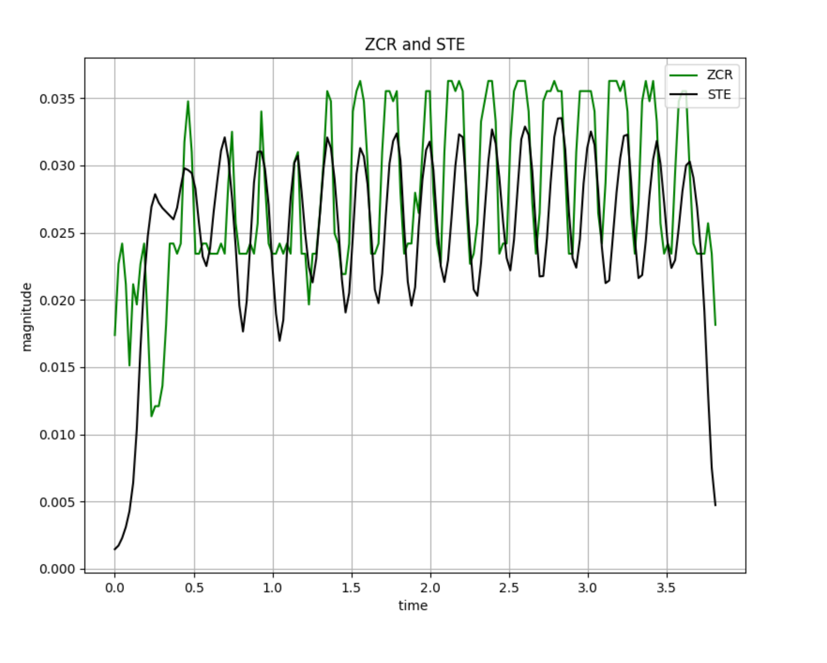

MFCC Coefficients
MFCC Coefficient Magnitude
time
MFCC 0
MFCC 1
MFCC 2
MFCC 3
MFCC 4
MFCC 5
MFCC 6
MFCC 7
MFCC 8
MFCC 9
MFCC 10
MFCC 11
MFCC 12

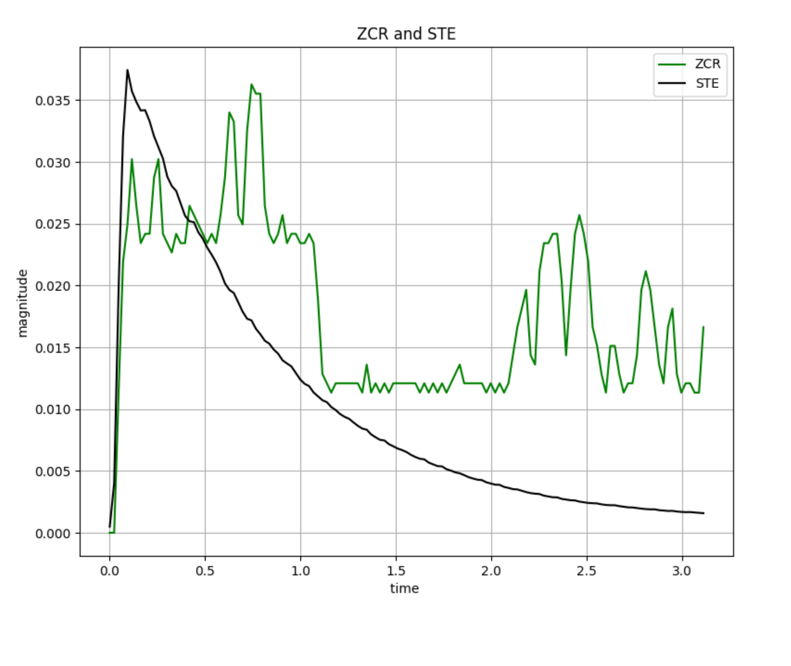

Spectral Contrast
Spectral Contrast 0
Spectral Contrast 1
Spectral Contrast 2
Spectral Contrast 3
Spectral Contrast 4
Spectral Contrast 5
Spectral Contrast 6
freq (Hz)
time

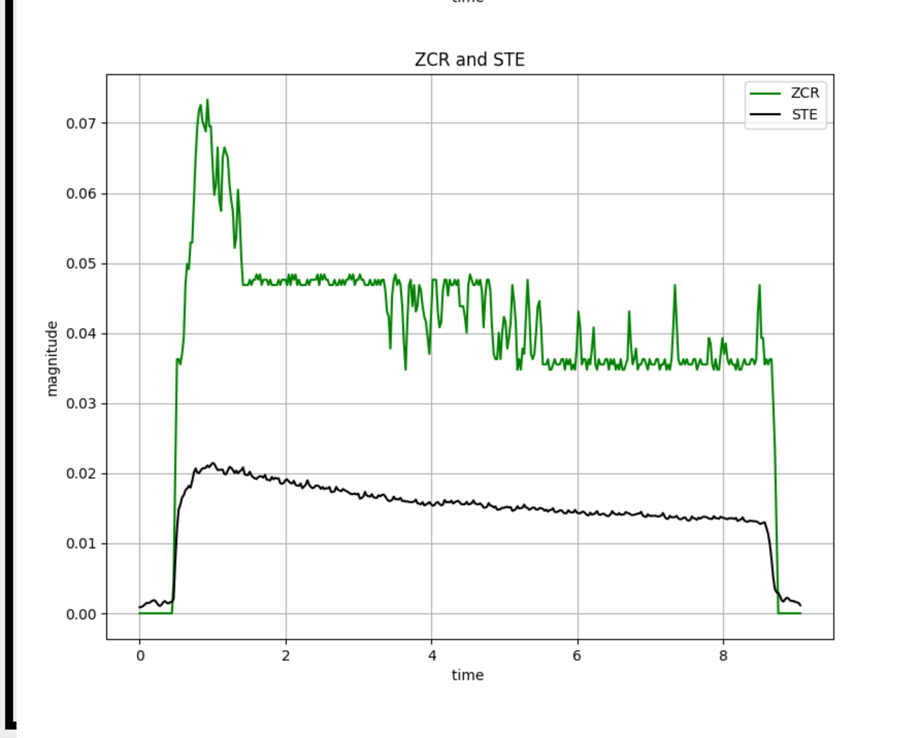

Spectral Contrast
Spectral Contrast 0
Spectral Contrast 1
Spectral Contrast 2
Spectral Contrast 3
Spectral Contrast 4
Spectral Contrast 5
Spectral Contrast 6
freq (Hz)
time

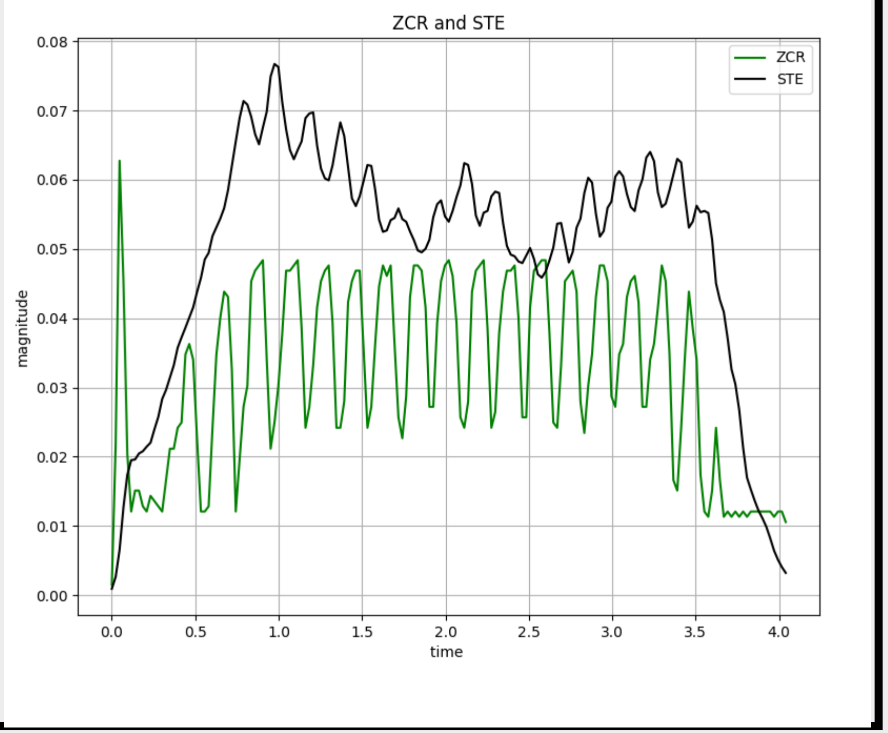

Spectral Contrast
Spectral Contrast 0
Spectral Contrast 1
Spectral Contrast 2
Spectral Contrast 3
Spectral Contrast 4
Spectral Contrast 5
Spectral Contrast 6
freq (Hz)
time


Spectral Contrast
Spectral Contrast 0
Spectral Contrast 1
Spectral Contrast 2
Spectral Contrast 3
Spectral Contrast 4
Spectral Contrast 5
Spectral Contrast 6
freq (Hz)
time


ZCR and STE
ZCR
STE
magnitude
time


ZCR and STE
ZCR
STE
magnitude
time


ZCR and STE
ZCR
STE
magnitude
time


ZCR and STE
ZCR
STE
magnitude
time