Automatic Music Transcription

Rishabh Aryan Das
rishabh.das@ufl.edu

PROJECT OBJECTIVES

- Attempt to present a supervised neural network model for music transcription
- Musical piece characteristics:
    - Has multiple musical sources
    - Each instrument piece is polyphonic ie, more than one note at a given time
- Motivation:
    - To make it easy for music beginners to learn to play the instrument effectively

DATASET

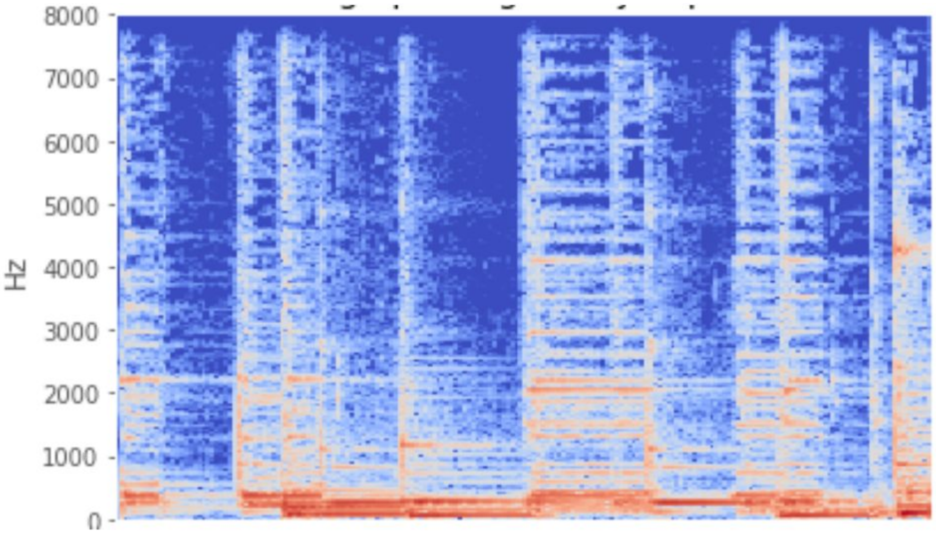
- MusicNet is the dataset we will use for training and evaluating the proposed models
- It contains train_data, test_labels, test_data and test_labels
- MusicNet has 330 groups each of which represents a different song. The names of the group are labelled by id_ID to identify a song
- Within each group there are two datasets, first one representing the audio signal while the second one represents all the labels of the song.
- The labels start_time and end_time are expressed in number of samples and so to obtain these labels in seconds we need to divide that number by the sampling rate (44100Hz)

INPUT REPRESENTATION

- A sound source is usually represented in time domain where the Y axis represents the amplitude and the X axis represents the time
- In order to distinguish the musical notes of the source sound we have to convert it from time domain to frequency domain
- This frequency domain is called **spectrum** which is the sum of a number of elementary cosine and sine signals of varying frequencies, amplitudes and phases
- Music audio signals are represented by discrete number of vectors
- We will use a Constant-Q Transform which applies a transform in logarithmic scale

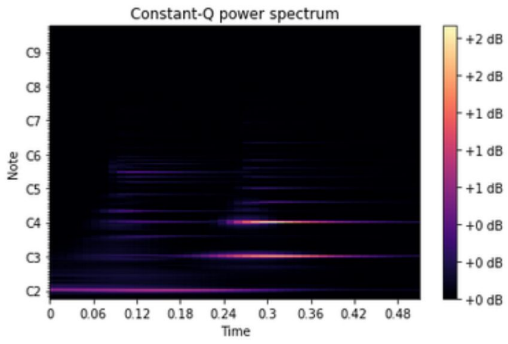
SPECTROGRAM

- Spectrograms are representation of the frequencies of acoustic signal with time
- It is a 3D matrix with time, frequency as the vertical axis and amplitude
- It becomes easier to distinguish the musical notes using spectrogram so the Neural Network can process them faster

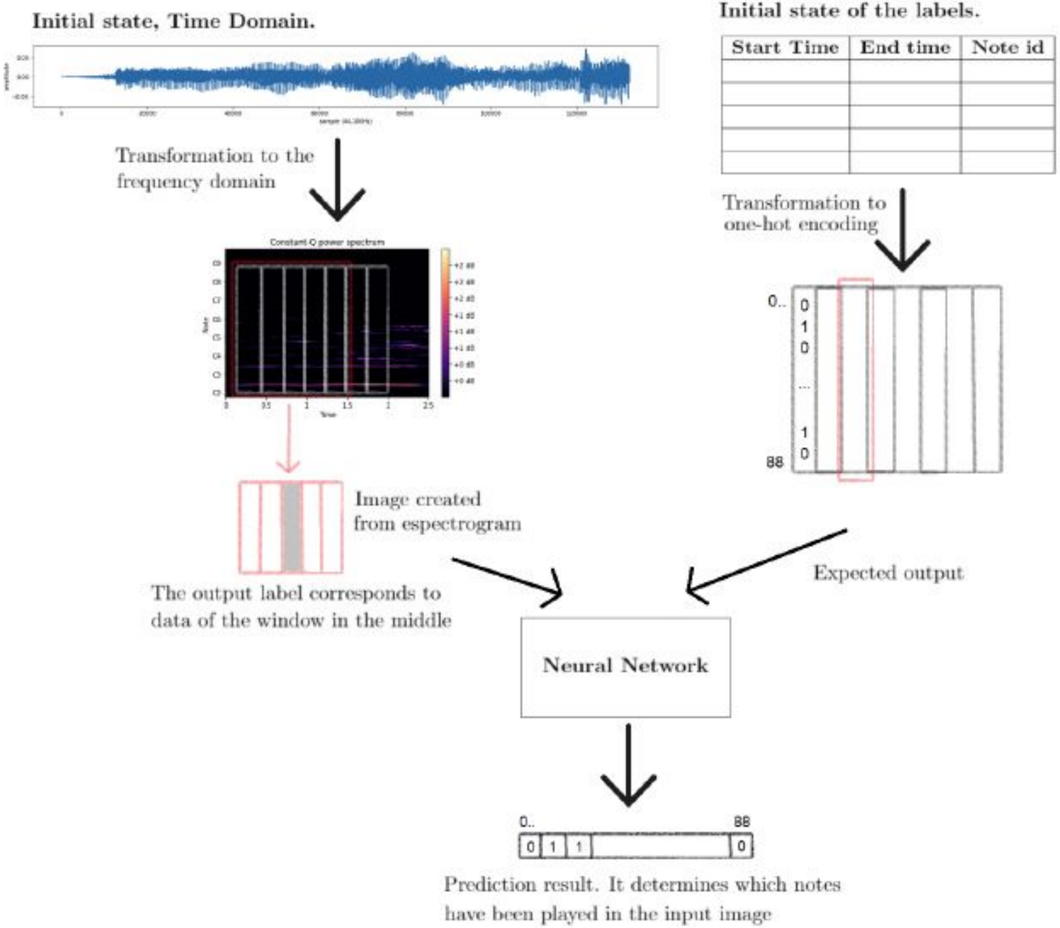


WHAT IS A CQT?

- Constant Q transform is a technique to calculate spectrogram using a logarithmic scale
- It transforms a time domain signal into frequency domain just like the FFT
- CQT is a bank of filters with geometrically spaced central frequencies
- CQT needs less memory space as compared to FFT which speeds up the training process
- CQT needs less number of values to represent a spectrogram



PROJECT PIPELINE

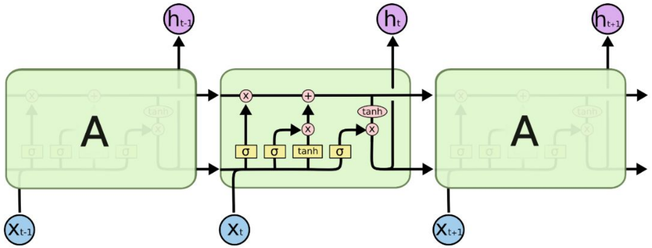


Concepts

- Deep Neural Networks (DNN)
  - Deep Neural Networks or DNNs are machine learning models used for linear and non linear classification and regression tasks.
  - Composed of several layers that are able to perform non linear transformations
- Recurrent Neural Networks (RNN)
  - RNNs were conceived as a solution for the inability of DNNs to handle sequential data
  - RNNs are suitable for AMT as the consecutive frames include both present and past features
- Long Short Time Memory Networks (LSTM)

What is Long Short-Time Memory Networks?

- LSTM are type of RNN architecture which can learn long term dependencies by using memory cell
- LSTMs can overcome the limitations of RNN to learn dependencies that are seperated several time steps in time due to vanishing gradient
- The memory cell does not use any activation function and the update step is done over 4 Neural Networks in each memory cell called gates.



Networks Structure

- Networks required are of two types DNNs and LSTMs.
- 4 hidden layers of DNNs will be used with 256 units in each hidden layer
- Stochastic Gradient Descent will be computed using Adam optimizer
- Activation function used in hidden layers is ReLU while Sigmoid activation is used for output layer as this layer is bounded by [0,1]
- The output layer is of size 88 units to represent all possible pitches
- The loss function is measured as the MSE between output and label vectors for each frame

Evaluation

- We use the sigmoid function in the output layer to round off the predictions
- To avoid overfitting after every epoch the validation set was evaluated to check the F measure.
- If the F measure did not improve for more than 15 epochs the training is halted.

$$\text{Precision}(P) = \sum_{t=0}^{N} \frac{TruePositives(t)}{TruePositives(t) + FalsePositives(t)}$$

$$\text{Recal}(R) = \sum_{t=0}^{N} \frac{TruePositives(t)}{TruePositives(t) + FalseNegatives(t)}$$

$$\text{Accuracy}(A) = \sum_{t=0}^{N} \frac{TruePositives(t)}{TruePositives(t) + FalsePositives(t) + FalseNegatives(t)},$$

$$\text{F-measure}(F) = \frac{2PR}{P + R}$$

Postprocessing

- To improve accuracy, instead of rounding the predictions we could train Hidden Markov Models
- Predicted pitches with duration smaller than the minimum duration of a pitch need to be removed