Student Nguyen Truong Phong

QuanTEEM 2023-2025

Signal processing project

**Spectrograms usage in machine-learning speech recognition algorithms**

**I. Project overview**

A spectrogram is a 2D representation of a 3D surface, where time is plotted on the x-axis, frequency on the y-axis, and the intensity of the signal (amplitude or power) is represented using color or grayscale. This layout is analogous to an image where pixels represent different values. We can say spectrograms are images of sounds, therefore, the voice recognition problem can be solved using image-processing techniques (CNN - Convolutional Neural Network).

This project studies how spectrogram can be used in a specific speech recognition problem – voice commands classification. We will build a CNN model to classify audio files into 8 classes. The silent class will be considered in the investigation of a real-time model. Also, the possibility of making a live model that classifies commands of users in real-time will be investigated.

**II. Data exploration**

Mini-speech commands dataset will be used. The dataset contains 8,000 files with length of approximately 1 second, separated equally into 8 classes ‘up’, ‘down’, ‘left’, ‘right’, ‘yes’, ‘no’, ‘stop’, ‘go’. Also, a supplement files of class ‘silence’ will be introduced to test the availability of the model in real-time classification.

Our data will be separated into 3 datasets: Train set, validation set and test set. There are no duplications inside and between these datasets for the accuracy of our evaluation. The accuracy of the model while training will be evaluated by the validation set. And the accuracy of the model predicting in the test set will be our final evaluation of the model.

A screen shot of a computer program

Description automatically generated

**III. Spectrogram Analysis**

For compatibility while processing data with the model, we use the method **‘tf.signal.sfft’**. This is the function in TensorFlow is used to compute the Short-Time Fourier Transform (STFT) of a signal. The STFT is a way to analyze how the frequency content of a signal changes over time. It breaks down a signal into its frequency components as they evolve in short, overlapping time windows. This function is particularly useful in tasks such as speech processing, audio analysis, and creating spectrograms for machine learning applications. The resulting tensor can be further processed to obtain magnitude spectrograms or other representations of the signal's frequency content over time.

A diagram of a sound wave

Description automatically generated

Fig. Short time Fourier Transform representation

In this project we will consider how different spectrogram processing before feeding to the model can affect the result of the model. The table below shows the effect of different spectrogram processing methods on the result. **Number of epochs** means the number of times the model has to train before reaching the best result.

One can conclude that using the normalized mel\_spectrogram provided the superior accuracy (**88.7%**) to other type of spectrogram processing. This result agree with The Impact of Audio Input Representations on Neural Network based Music Transcription [1] <https://arxiv.org/pdf/2001.09989.pdf> where Mel\_spectrogram is the best choice for audio preprocessing.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Spectrogram Analysis method | Accuracy (%) | No. of epochs | Time (s) | Time per epoch (s) |
| Base spectrogram function | **78.8** | 26 | 342 | 13.2 |
| Using log scale spectrogram | **82.5** | 18 | 232 | 12.9 |
| Normalized mel\_spectrogram | **88.7** | 16 | 331 | 20.7 |

Table.1 Result of different spectrogram processing approaches to audio files

A collage of images of a blue screen

Description automatically generated with medium confidence

Fig . Spectrogram using the spectrogram base function only

A collage of images of a heat wave

Description automatically generated

Fig . Spectrogram using the log scale spectrogram

A collage of images of a sound wave

Description automatically generated

Fig Normalized\_mel\_spectrogram