Midway UrbanSound Classification P10

Jay Jagtap	Rohan Pillai	Sri Harsha Varma	Sai Shashank N
jjjagtap@ncsu.edu	rspillai@ncsu.edu	suppala@ncsu.edu	snayabu@ncsu.edu

1 Background & Introduction

Classification of images is a very popular research area with several applications in information classification, indexing and retrieval. However, there isn't much research in a related area of audio classification particularly music, speech or sounds classification. Some of the current research in this area is only restricted to identification of auditory scene type. There isn't any research in reliable identification of the source of sound such as an idle engine, chirping of birds, etc [1]. Lack of reliable and labelled audio data could be attributed for this lack of any research in this domain. Lack of a common vocabulary makes it even more difficult for comparing results since the classification results vary from study to study.

This paper proposes a classification of urban sounds that is based on the taxonomy of sounds introduced in [2]. Any such taxonomy should satisfy three requirements in order to successfully address the issues stated above. Firstly, it should take into consideration any previous such taxonomy or research. Secondly, it should strive to contain as many low-level details as possible. Lastly, it should focus on sounds that contribute to urban noise pollution. To make sure this paper follows these requirements, we will be following the taxonomy introduced in [3] contributed to urban acoustics. We will be providing low-level details such as "horn", "engine", "brakes", etc.

The dataset contains 8,732 labelled sound samples (less than 4s) classified into 10 classes- Air Conditioner, Car Horn, Children Playing, Dog Bark, Drilling, Engine Idling, Gun Shot, Jackhammer, Siren, and Street Music. The dataset follows the taxonomy described above. These sound excerpts are digital audio files in .wav format. Sound waves are digitised by sampling them at discrete intervals known as the sampling rate. The excerpts are sorted into 10-folds and were taken from field recordings uploaded to `www.freesound.org`.

2 Method

The audio files in the folders(fold1 - fold10) consists of a representative sample(i.e. files equally distributed from all the sources under consideration) and thus help in the 10-cross validation.

In addition to the audio files, a CSV file which consists of the meta data of the the audio files has been provided. The meta data has various attributes: slicefilename, fsID, start, end, salience, fold, classID, class

slicefilename: The name of the audio file.

fsID: The Freesound ID of the recording from which this excerpt (slice) is taken.

start: The start time of the slice in the original Freesound recording.

end: The end time of slice in the original Freesound recording.

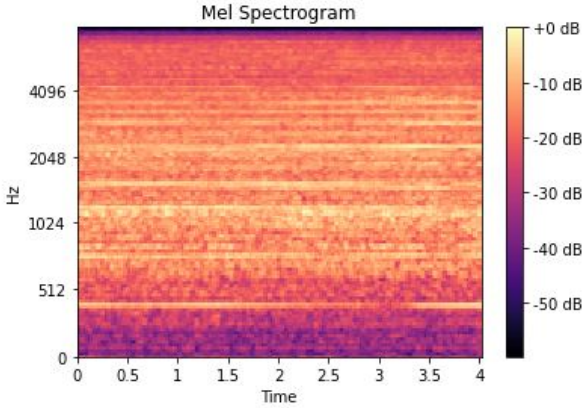
salience: A (subjective) salience rating of the sound. 1 = foreground, 2 = background.

fold: The fold number (1-10) to which this file has been allocated.

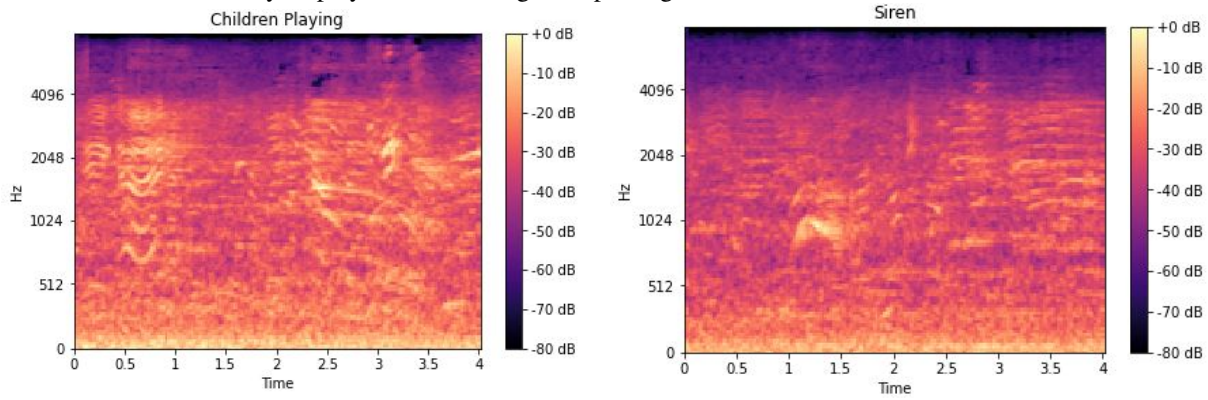
classID: A numeric identifier of the sound class: 0 = airconditioner 1 = carhorn 2 = childrenplaying 3 = dogbark 4 = drilling 5 = engineidling 6 = gunshot 7 = jackhammer 8 = siren 9 = street music

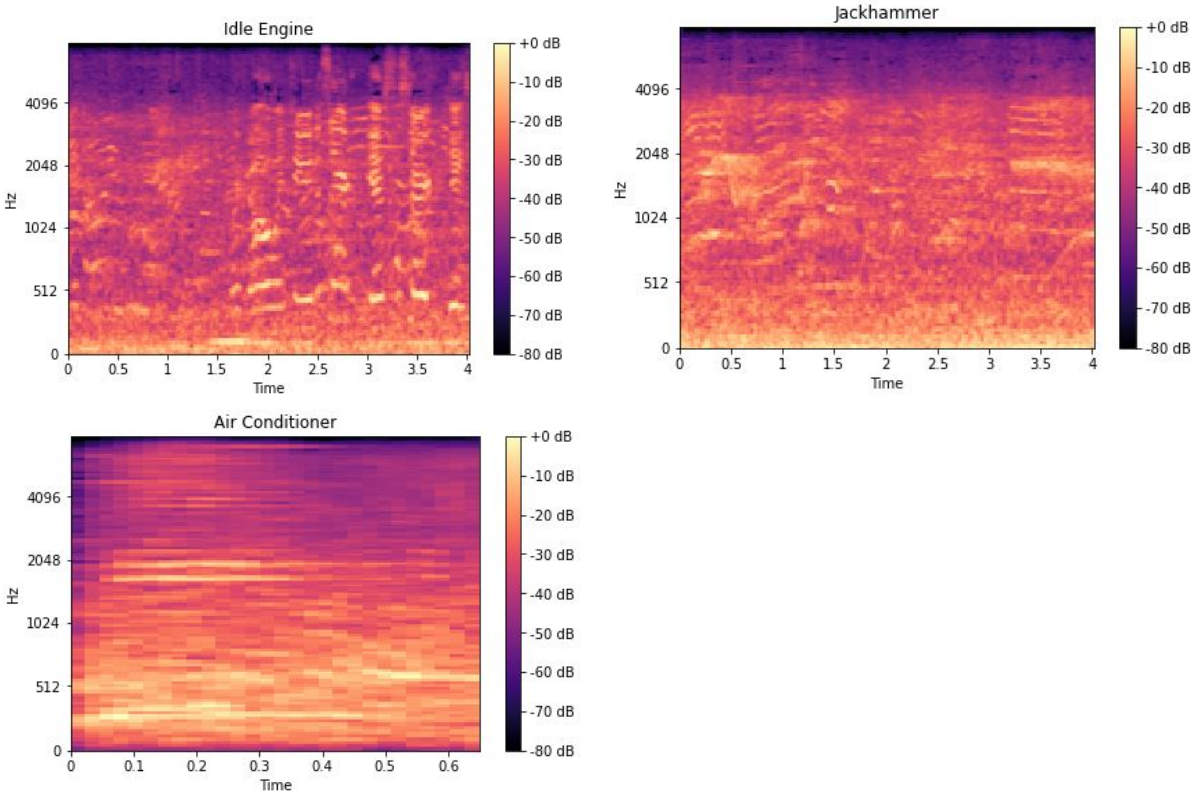
class: The class name: airconditioner, carhorn, childrenplaying, dogbark, drilling, engineidling, gunshot, jackhammer, siren, street music.

To deal with audio files, we have used the publicly available Python library librosa which is a package for music and audio analysis. It has basic functionalities such as loading and pre-processing the audio files. When we load any audio file using librosa, it returns a tuple with two objects. The first item is an audio time series array for the audio file, and the second object is its corresponding sampling rate. The following plot shows two dimensional representation of a random audio file in the data set:
After loading an audio file, we now need to create a mel spectrogram for it. We cannot plot the frequencies at their present scale because humans cannot perceive frequency differences at a linear scale. For instance, humans can identify the difference 500 Hz and 700 Hz but will not be able to tell the difference between 10k Hz and 11k Hz. So for this purpose, we first need to change the scale accordingly. This scale is called the mel scale [4]. A mel spectrogram is a spectrogram where the frequencies are converted to the mel scale. Below is a mel spectrogram for the above audio file.



Creation of the spectrogram or wave-plot is often the prerequisite for the sound classification neural network. Spectrograms are a useful technique for visualising the spectrum of frequencies of a sound and how they vary during a very short period of time. A mel is a number that is indicative of the pitch of the audio file. Librosa is capable of extracting the power spectrogram for each mel over time as well as a function for easy display of the resulting mel spectrogram.





After extracting the data .wav files and stacking it in feature numpy arrays, we feed it to convolutional Neural Network for classification. Convolutional Neural Network perform better with image classification due to their feature extraction and classification parts. During this stage of our project, we have used Neural Network Architecture. Our Neural Network has 2 convolution layers, each convolution followed by a max pooling layer to extract dominant features. We also add a dropout layer to reduce reliance on a single feature. Each of the convolution layer has a tanh activation function and we have a softmax layer at the end to get the classification probability vector.

Following is the model summary:

```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d (Conv2D)              (None, 16, 8, 64)         640
_________________________________________________________________
max_pooling2d (MaxPooling2D) (None, 8, 4, 64)          0
_________________________________________________________________
conv2d_1 (Conv2D)            (None, 8, 4, 128)         73856
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 4, 2, 128)         0
_________________________________________________________________
dropout (Dropout)            (None, 4, 2, 128)         0
_________________________________________________________________
flatten (Flatten)            (None, 1024)              0
_________________________________________________________________
dense (Dense)                (None, 1024)              1049600
_________________________________________________________________
dense_1 (Dense)              (None, 10)                10250
=================================================================
Total params: 1,134,346
Trainable params: 1,134,346
Non-trainable params: 0
_________________________________________________________________
```

3 Experiment Setup

Our data-set is divided into 10 folds for classification. This enables us to do k-cross validation and have a better measure of our accuracy. The tools we have used are python, numpy, librosa library for audio processing and tensorflow for fitting data to Neural Networks. We have leveraged Google Cloud Platform for training purposes.

4 Results

Following are the accuracies of our 10-cross validation on the data set.

Fold Number	Accuracy
1	54.36
2	62.18
3	56.98
4	68.44
5	48.22
6	66.24
7	72.34
8	46.76
9	51.03
10	48.97
Average	57.5

5 Conclusion

The average accuracy after 10 cross-validation is 57.5%. In the next phase of our project, we are aiming for an accuracy of more than 80%. To do so, we plan on leveraging state of art neural networks by using transfer learning to improve our CNN as well as explore and implement other machine learning approaches.

References

[1] Chu, S., Narayanan, S. & Kuo C.C. (1995) Environmental sound recognition with time-frequency audio features., *IEEE TASLP*, 17(6):1142–1158, 2009.

[2] Salamon, J., Jacoby, C., & Bello, J.B. (2014) A Dataset and Taxonomy for Urban Sound Research. *In Proceedings of the 22nd ACM international conference on Multimedia (MM '14). Association for Computing Machinery New York, NY, USA, 1041–1044.* DOI:https://doi.org/10.1145/2647868.2655045

[3] Brown, A.L., Kang, J., & Gjestland, T. (2011) Towards standardization in soundscape preference assessment. Applied Acoustics, 72(6):387–392.

[4] Stevens, S., Volkmann, J., Newman, E.B. (1937). A Scale for the Measurement of the Psychological Magnitude Pitch.