International Journal of Current Trends in Engineering & Technology
www.ijctet.org, ISSN: 2395-3152
Volume: 07, Issue: 04 (July-August, 2021)

Classification of Urban Sound using Convolutional Neural Network

Ankit Ojha, Akarsh Dwivedi, Aditya Prakash
Department of Information Technology
Galgotia college of engineering and technology Greater Noida, India
ankit2705ojha@gmail.com, akarshdwivedi99@gmail.com, prakash.aditya36@gmail.com

Abstract—This research paper describes our experience in creating Urban sound classification, which is currently a growing area of research where numerous real-world applications could be found. Many different research-related works are happening in audio fields, such as speech and music recognition, but the classification of environmental sounds is comparatively scarce. This project used a Convolutional Neural Network (CNN) as an Image classifier where the Spectrogram image of noise is used. CNN is very effective in image classification and classifying audio noises by converting them into visible spectro images. Our eyes capture visuals while our ears detect sounds. So Basically, eyes were far more superior and faster when compared to ears; hence we are using visible classifiers.

Keywords— Urban noise recognition · Audio Waveform· Convolutional Neural Network · Mel-Frequency Cepstral Coefficients.

I. INTRODUCTION

This project is inspired by the replication of information processing by the brain using mathematical modelling and simulation. The theoretical understanding of neuroscience helps create the basic algorithms or models used in several Neural Networks/ Computer Vision areas. "Computational neuroscience" is the whole structure of thinking and processing information in our brain. It explains the biological mechanisms of processing in neurons, and computers simulate these neural circuits, implementing them in Matlab, Python. We Used a Convolutional Neural Network (CNN) model as an Image classifier that needed a Spectrogram image of noise. Since CNN is very effective in image classification, we used it to classify audio noises into visible spectro images. Our eyes are generally better than our ears in sensing. So Basically, eyes are faster in classifying things when compared to ears, so we conclude by using visible classifiers. This project aims to replicate a function of the Brain, which is to identify and differentiate different kinds of sounds and noises. Hence, we build a learning model which can detect random noises just like the brain.

II. BLUEPRINT

During the 1950s, David Hubel and Torsten Wiesel proposed how animals perceive their environment by experimenting on the brain functionality in mammals. The vision research has been going on since the 1950s. This research finally gives us a very powerful algorithm known as the Convolutional Neural Network (CNN). CNN is very capable of performing Machine Learning classification on images as it is very well developed. Every day, new advancements come up, showing remarkable improvement in the algorithm or some new use case. In image classification, where convolutional neural networks are used to classify images with extremely high accuracy, this technology can be used in other domains, such as sound classification.

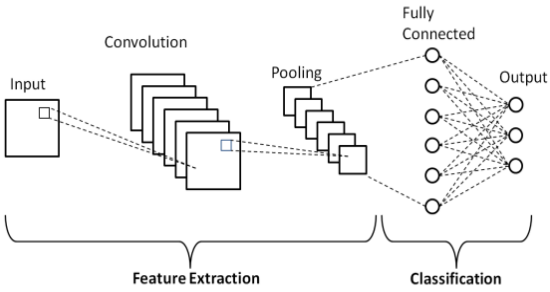


Fig.1 CNN Layers

A Convolutional Neural Network is a Deep Learning algorithm that takes required images, assigns some properties (learnable criteria) to various aspects of the image, and differentiates one from the other. For this system, we used a dataset of noises like:

- Air Conditioner
- Car Horn
- Dog barking
- Drilling
- Cars in traffic
- Children Playing

The next step is extracting the features which are needed to train our model. After this, a diagrammatic representation of each audio file is created to identify classification features. "Spectrograms" are a process for visualising the pictorial spectrum of frequencies of a

sound and how they alter during a period. Mel-Frequency Cepstral Coefficients (MFCC) is a very known technique for this process. The Convolutional Neural Network processes the MFCC Spectrometer image based on different classifications like frequency, pitch, direction.

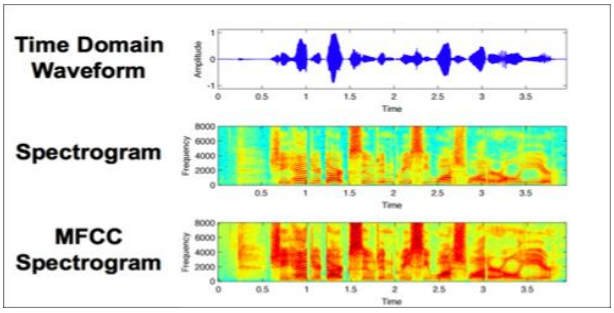


Fig.2 Waveform to MFCC Spectrogram

In this project, we used software, which are:

- Librosa's mfcc() codec (for the waveform to spectrogram conversion)
- Librosa (Python package for music and audio processing)
- Spyder (Python IDE)
- Keras Conv2D Architecture (for creating CNN in python)
- Windows 10 Sound Recorder (for capturing sounds)

III. METHODOLOGY

For this system, we used a dataset of 6 random noises. These digital audio files that we have taken are in .wav format. After that, we identify these audio properties that are retrieved and process them to ensure consistency across the whole dataset. The consistency factors are:

• Audio Channels
• Sample rate
• Bit-depth

In this process, we create a visual graph of each of the audio samples so that we can identify features for its classification.

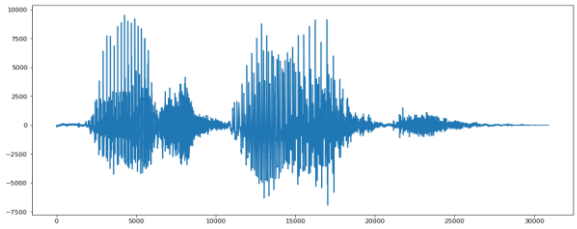


Fig.3 Waveform Sound

"Librosa", a Python package created by Brian McFee, is used in processing audio files and allows us to load audio clips in our Python IDE for analysis and manipulation. To analyse the spectrogram, we divide the audio clip into small millisecond clips and compute Short-Time Fourier Transform (STFT) on each divided clip. Then, we plot these short clips as a coloured vertical line in the spectrogram. Each audio file has to extract an MFCC image (meaning having diagrammatic representation for each audio file) and store these images in the Panda Data frame along with its classification label. Librosa's mfcc() function is best suited to generate an MFCC image from audio data. Then, we build our model based on the Convolutional Neural Network (CNN) using Keras Conv2D and a TensorFlow backend in Python IDE. The final output layer in our project has 10 nodes used to match the total number of possible classifications.

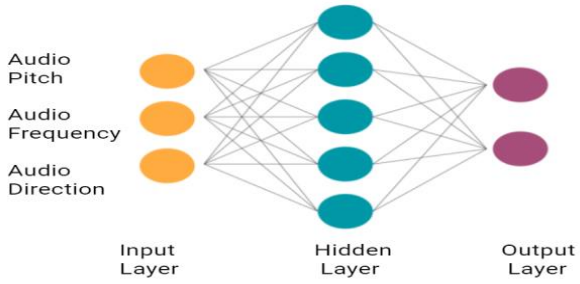
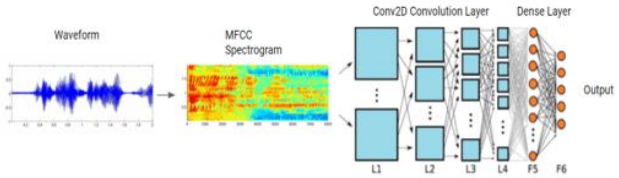


Fig.4 CNN sound Classification

In our model, we used these three parameters for compiling our model :

• Loss function - This is the most common choice for classification. A lower loss function score means the model is performing better. This parameter used categorical cross-entropy.
• Metrics - Accuracy metrics give us the accuracy score and validate the data when we train the model.
• Optimizer - we used Adam, which is generally a good parameter for many use cases.

We build and train a Deep Neural Network model with these data collected and make predictions on it. CNN is typically a good classifier that performs particularly well with image classification tasks due to its feature extraction and classification parts. We believe that this feature effectively finds patterns on MFCC, much like finding patterns within images. We start with 100 epochs which is the number of times the model cycle through the data. This model keeps improving itself on each cycle until it reaches its best accuracy. Test with sample data. We verify the predictions using a subsection of the sample audio files we explored previously. We expect the bulk of these to be classified correctly.



Fig, 5 MFCC Spectrogram to Conv2D layer

IV. CONCLUSION

Table I. Comparison of Testing accuracies

S. No.	Model	Accuracy Percentage
1.	Pre-training	12%
2.	1st training model	76%
3.	2nd training model	84%
4.	After additional refinements	92%

Our final model achieved a classification accuracy of 92% using our audio files. The final model performs very well when presented with a random .wav file for a few seconds, which returns a reliable result. However, we do not know how our model would perform on Real-time audio. Also, we don't know how our classifier would perform in a real setting. Our study makes no attempts in taking other factors such as noise, echo, volume and salience level of the sample. If we were to continue with this project, some additional areas could be explored:

- As previously mentioned, test the models' performance with Real-time audio.
- Experiment to see if per-class accuracy is affected by using training data of different durations.
- Experiment with other techniques for feature extraction, such as different forms of Spectrograms.

ACKNOWLEDGMENT

We would also like to show our gratitude to Dr Sunil Kumar Bharti, Professor, Galgotia college of engineering and technology, for sharing their pearls of wisdom with us during this research, and we thank our friends who help in collecting resources for this research. We are also immensely grateful to Mr Prem Prakash for their comment on an earlier version of our model, although any errors are our own and should not tarnish the reputations of these esteemed persons.

REFERENCES

[1].Neisser, U. (1967). Cognitive Psychology. New York: Appleton-Century-Crofts.
[2].Chachada, S., & Kuo, C. (2014). Environmental sound recognition: A survey. APSIPA Transactions on Signal and Information Processing 1.
[3].Deep learning in neural networks: An overview by "Jürgen Schmidhuber."
[4].Udacity Capstone Project on Machine Learning Nanodegree 2018 by "Mike Smales".
[5].Yu, C. Y., Liu, H., & Qi, Z. M. (2017). Sound event detection using deep random forest. In Detection and Classification of Acoustic Scenes and Events.
[6].Piczak, K. J. (2015, September). Environmental sound classification with convolutional neural networks. 2015 IEEE 25th International Workshop on Machine Learning for Signal Processing (MLSP).
[7].Artificial Neural Network (ANN) By Jake Frankenfield at Investopedia.
[8].Salamon, J., & Bello, J. P. (2017). Deep convolutional neural networks and data augmentation for environmental sound classification. IEEE Signal Processing Letters.
[9].Qiao, T., Zhang, S., Zhang, Z., Cao, S., & Xu, S. (2019). Spectrogram Segmentation for Environmental Sound Classification via Convolutional Neural Network and ScoreLevel Fusion.