Land Use Prediction based on Auditory Datasets

Abhishek Kuriyal[1], Harshul Raj Surana[2] and Shraddha Aggarwal[3]

Abstract— **This final report presents a low-complexity CNN based deep learning framework for predicting usage of land in different settings namely, Residential, Industrial and Commercial. The proposed architecture constitute of two main steps front-end feature extraction from images and back-end neural network. Since, the input data is in audio format, it has to be converted into spectrogram images that correctly represents the frequency vs time representation of audio wave. Gammatone spectrogram representation is employed as front-end feature extraction in our framework. Spectrogram thus extracted are fed into CNN-based architecture for classification. All submission models were trained with manually collected data from multiple sources and results obtained were valid for test samples from various areas of Bhopal city.**

I. INTRODUCTION

Due to recent advancements in processing power and computational technology such as GPU (Graphic Processing Unit) and TPU (Tensorflow Processing Unit) along with increase in data collection methodologies, deep learning has become more and more feasible. Increasing development in deep learning has provided powerful techniques for various disciplines such as computer vision, natural language processing and recently emerging research field named ”machine hearing” [1]. In [2], authors have showed the robustness of CNN-based architectures. In some cases architectures performed well enough to surpass performance in human adept tasks like classification and identification. However, it is a general trend that these models demonstrate remarkable performance but with a heavy computational cost and large number of parameters. This is presented in [3] where number of parameters corresponding to some famous models are demonstrated in tabular manner. Within this report we propose a deep learning framework with low complexity CNN-based model and compare it with other widely used models.

II. CONTRIBUTIONS

Our model poses two significant contributions:-

1. Created a neural network model that performs well over a large range of input samples and does not generate absurd amount of hyper-parameters (less than 1M for approximately 10,000 input training samples). CNN models have shown exemplary performance in training image data. Thus, audio first needs to be converted into a visual representation. Authors in [4] have described some of the algorithms and techniques to make this conversion possible. It is however, important that the converted image retains the essential features of audio such as frequency variation with time, sound decay, loudness, audio quality etc. More so if the image captures how the audio is perceived by human ears it will result in better input data for classification. We have demonstrated how such data can be prepared by employing Gammatone filter banks.

2. Created an interface for a fast classifier that can produce images with an execution time less than 5 seconds on average. This can be directly installed in a drone based environment for real time prediction. Considering the complexity of the task it is important that large amount of data is fed into this model to improve the prediction accuracy.

III. BACKGROUND

Acoustic Scene Classification is the classification of an audio signal into one of the provided pre-defined classes that characterizes that environment,

[1] Abhishek Kuriyal, 19009, Department of Data Science and Engineering

[2] Harshul Raj Surana, 19294, Department of Data Science and Engineering

[3] Shraddha Aggarwal, 19137, Department of Data Science and Engineering

is a broad investigation area related to computational auditory scene analysis (CASA.). Acoustic scene classification (ASC) is based on the analysis of the audio signal recorded at the scene, under the assumption that the general acoustic characteristics of the scenes are recognizable. State-of-the-art solutions are based on spectral features, most commonly the log-mel spectrogram, and convolutional neural network (CNN) architectures, often used in large ensembles.

Zoning is a method of urban planning in which a municipality or other tier of government divides land into areas called zones, each of which has a set of regulations for new development that differs from other zones. Zones may be defined for a single use (e.g. residential, industrial), they may combine several compatible activities by use, or in the case of form-based zoning, the differing regulations may govern the density, size and shape of allowed buildings whatever their use.

The most commonly and globally single-use zone defined is Residential, Commercial and Industrial.

ASC has been a part of the Detection and Classification of Acoustic Scenes and Events (DCASE) challenge since its inception in 2013. In the first three editions of the challenge, the acoustic scene classification task has received the highest number of submissions among the available tasks, with 17 submissions in 2013, 48 submissions in 2016, and 97 submissions in 2017.

The class labels of the Dataset are of specific areas, for example office, urban park, etc. The latest DCASE 2021 had a subclass asking for prediction on Indoor, Outdoor and Travel. This is the closest to our 3-classed classification task.

DCASE challenge is one of the competitions that is active on related fields to computational analysis of sound events and scene analysis, namely acoustic scene classification, sound event detection, and audio tagging. It started in 2013 including two tasks and 18 international participants, and has grown to already 87 teams in 2016 with 4 different tasks. Another benefit of competitions such as DCASE is to provide a fair comparison, and a baseline dataset for researchers to work with, and make the results of their studies more concrete for others to compare with

However, all the classification tasks done previously employs heavy models with large number of parameters which motivated us to search for a lower complexity model that could produce acceptable results in real time.

IV. MATERIAL AND METHODS

Entire methodology can be explained in multiple subsections.

A. Data Preparation and Cleaning

The training dataset used in the project had been prepared using a combination of YouTube videos and publicly available web sources. The YouTube videos were first converted into mp3 format and then to wav format to achieve higher sound quality using online audio converters. After converting them into wav format, we split the audios into the clips of length 10 seconds. Similar length audio clips is must in order to create uniform size spectrogram images. During the splitting process, many audio clips contained no information or the loudness was extremely low. These were then discarded and only the clips with sufficient information were chosen. Several online datasets included recordings of length less than 10 seconds which were discarded and those that were longer than 10 seconds were split into the clips of length 10 seconds. We then manually classified all of the samples into three different classes, namely Residential, Commercial and Industrial Areas.

In order to prepare the testing data, we manually recorded noises in several cities across India, including Bareilly, Mumbai, Pune, Bangalore and primarily different locations of Bhopal city for the three classifications, for example- IISER Bhopal Campus for Residential area, Bairagarh for Commercial area and Govindpura for Industrial area. Again, we converted these recordings into wav format, split them into the clips of 10 seconds and then cleaned the testing data.
Below are some sources that were readily available and were utilised to prepare the training dataset:-

https://zenodo.org/record/3819968.YZt5YNBBxPZ
https://urbansounddataset.weebly.com/
urbansound8k.html

http://dcase.community/challenge2020/task-acoustic-scene-classification

Following are the YouTube links of videos that were utilised in the project:-

For Residential Areas:
https://youtu.be/7iDxLF2PWFw
https://youtu.be/gvNmkJnCjdU
https://youtu.be/wAjKpdokhls
https://youtu.be/7kO80A7CDWc
https://youtu.be/n529hDCIoAE
https://youtu.be/7vXiXIlKTFc
https://youtu.be/J1l8RMR-qXM
https://youtu.be/5tX71WGrs4
https://youtu.be/UZ9uyQI3pF0
https://youtu.be/rYoZgpAEkFs
https://youtu.be/omxX-0PjXkY

For Commercial Areas:-
https://youtu.be/w7YksRIVgc
https://youtu.be/WoxnL5dakyA
https://youtu.be/GLh0w4XiAHw
https://youtu.be/vHEIImSM1Do
https://youtu.be/ReBAbY5JVZs
https://youtu.be/vHEIImSM1Do
https://youtu.be/0fqt2AMpTg0
https://youtu.be/QJsjWELzq-Q
https://youtu.be/zQyhILUvMMM
https://youtu.be/RHOCIpB8ac0
https://youtu.be/x2UulCWGess
https://youtu.be/G2Hrh4daro

For Industrial Areas:
https://youtu.be/CSZxwFfwyyg
https://youtu.be/rU0v9uTwtUY
https://youtu.be/xHc8Y37ApZg
https://youtu.be/UjnFRp7vtVM
https://youtu.be/1jqlrofW88
https://youtu.be/eKA2CA5mOOw
https://youtu.be/uyc4u6dh4Cs
https://youtu.be/2kVfqNfblQ
https://youtu.be/bDZVYIhIEs
https://youtu.be/hnIOZQWzTO0
https://youtu.be/kxHmGn50nmk

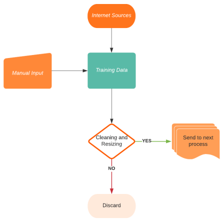


FIG 1: Complete data extraction and cleaning process

B. Preprocessing

The input audio needs to be converted into spectrogram images before feeding into the CNN. We chose Gammatone spectrogram for following reasons:- for several years, MFCC (Mel-frequency cepstral coefficient) was used in many of the audio-image conversion and Automatic Speech Recognition (ASR) tasks. however in [5], the contrasts of MFCC and GFCC(Gammatone frequency cepstral coefficient) have been compared. GFCC have showcased to be a significant improvement in noise robust audio samples.

Gammatone filters are linear approximations of physiologically motivated processing performed by the cochlea of our ears. The main purpose of this filter is to model human auditory systems and consists of a series of bandpass filters. Filters are defined by the following impulse response:

$$f(t) = at^{n-1}cos(2\pi f_c t + \phi)e^{-2\pi bt}$$

where n is the order of the filter, b is the bandwidth of the filter, a is the amplitude, f_c is the filter center frequency, and ϕ is the phase.

In [6] Glasberg and Moore relate center frequency and the ERB(Equivalent rectangular bandwidth) of an auditory filter as

$$ERB(f_c) = 24.7((4.37 f_c)/1000 + 1)$$

The output of the mth gammatone filter, X_m can be expressed by

$$X_m = x(k) * h_m(k),$$

where $h_m(k)$ is the impulse response of the filter.

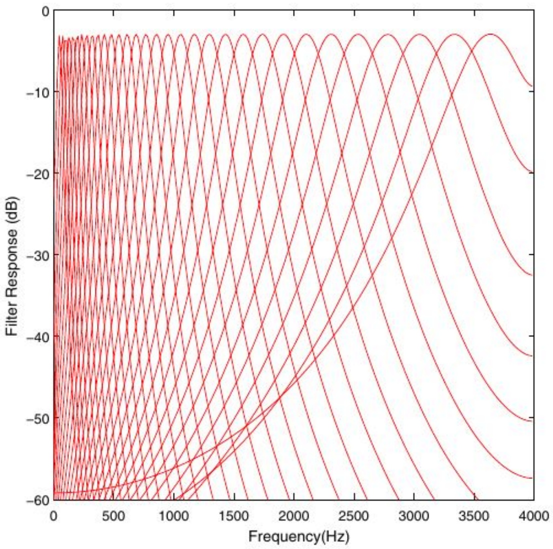


FIG 2: Frequency response for the 32-channel gammatone filter bank.

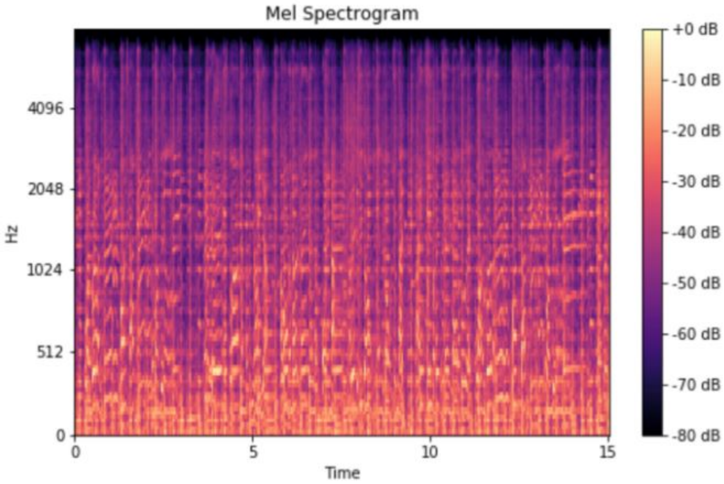
Modulation Spectrum:- The long-term modulations examine the slow temporal evolution of speech energy with time windows in the range from 160 to 800 ms, contrary to the short-term modulations with time modulations of 10 to 30 ms. Short term modulations capture the rapid changes of the speech signals.

Improving results of spectrogram filtering:-

On observing the modulation spectrum of audio samples, we were able to identify some interesting phenomena. modulation spectrum $Y_m(f,g)$ is obtained by applying Fourier transform on the spectra, and is obtained by taking absolute values $|Y(t,f)|$ at each frequency where $Y(t,f)$ is the time frequency representation after short time Fourier analysis, expressed as,

$$Y_m(f,g) = FT[|Y(t,f)|]|_{t=1,..T}$$

Where T is the total number of frames, and g is the modulation frequency. It has been observed that for noisy environments, the components of the modulation spectrum below 2Hz and above 10Hz are less important for speech intelligibility especially band below 1 Hz which contains mostly information about the environment. Thus we can extract these components to get even finer details about the environment. For a scenario of a person speaking, these components can be used to predict the environment surrounding that person. Below figures demonstrate the difference between mel-frequency based spectrogram and gammatone filter based spectrogram.



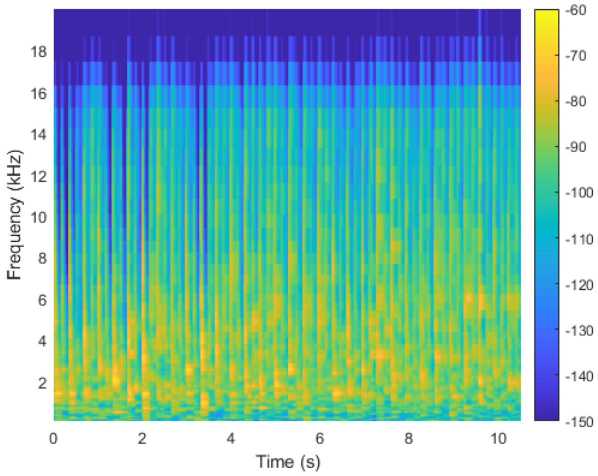


FIG 4: Gammatone based Spectrogram

C. Neural Network Architecture

For this model, we proposed a Convolutional Neural Network (CNN) based architecture. The name follows from the ability of these networks to perform “Convolutions” which are feature extraction operations that identify the most prominent components of an image like border, shape, edge and color distribution etc. Deep Convolutional Neural Network(DCNN) [7] is a special type of CNN, which has demonstrated exemplary performance in image processing tasks. The powerful learning ability of deep CNN is primarily due to the use of multiple feature extraction stages that can automatically learn representations from the data.

Our model uses a non-complex DCNN with only 5 hidden layers. Despite that our model performs fairly well to extract the necessary features it requires from corresponding training images. Choice of good sets of training parameters like dimensions and number of filters, strides, padding etc can result in greater accuracy. Contrary to this, having a large number of hidden layers does not necessarily imply the latter. The hidden layers can be divided into separate parts:

Convolutional layer:- Convolutional layer is composed of a set of convolutional kernels where each neuron acts as a kernel. But if the kernel is symmetric, the convolution operation becomes a correlation operation (Ian Goodfellow et al. 2017). Kernel works by dividing images into small blocks and extracting feature motifs. Elements of kernel are then multiplied with the corresponding elements of the receptive field Convolution operation can be expressed mathematically :-

$$f_l^k(p,q) = \sum_c \sum_{x,y} i_c(x,y) e_l^k(u,v)$$

where, $i_c(x,y)$ is an element of the input image tensor I_C, which is element wise multiplied by $e_l^k(u,v)$ index of the k^{th} convolutional kernel k_l of the l^{th} layer.

Convolution is computationally expensive process. Time taken to perform entire convolution is proportional to size of input matrix and number of kernels/convolution filters. In our model, only 3 convolutional layers were used.

(b) Pooling layer:- Once features are extracted, its exact location becomes less important as long as its approximate position relative to others is preserved. Pooling or down-sampling is an interesting local operation. It sums up similar information in the neighborhood of the receptive field and outputs the dominant response within this local region (Lee et al. 2016).

$$Z_l^k = f_p(F_l^k)$$

Above equation shows the pooling operation in which Z_l^k represents the pooled feature-map of l^{th} layer for k^{th} input feature-map F_l^k , whereas $f_p(.)$ defines the type of pooling operation.

Pooling operation can help to extract combination of features, which are invariant to translational shifts and small distortions. Pooling is fairly straightforward operation and it does not generate any trainable parameters. The operation depends entirely on size of filters/kernels/weights used and the stride used.

(c) Activation Function:- Activation function helps in learning of intricate patterns. Using appropriate activation function can accelerate the learning process significantly. Different activation functions can be used depending on the type of classification problem. In most cases, tanh, sigmoid, maxout, SWISH, ReLU and the variants of ReLU, such as leaky ReLU are used. In our model we employed ReLU activation function for all hidden layers, except the dense layer which uses Softmax activation function.

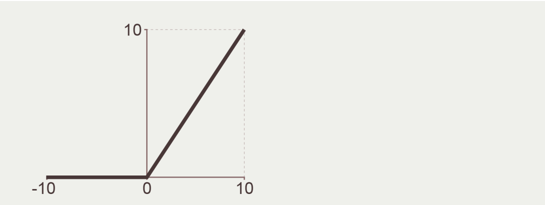


FIG 5: ReLU activation function

(d) Fully Connected Layer:- Mostly used at the end of the CNN for classification. The name follows from the fact that all the elements of input matrix is connected to all the features of the output matrix. Fully connected layer is important because it takes input from feature extraction and globally analyses the output of all the preceding layers (Lin et al. 2013). The output from fully connected layer passed into subsequent fully connected layer. The final layer which utilizes SoftMax as the activation function converts fully connected layer into output similar to dimension of number of classes of prediction.

In our model we utilized only 1 fully connected layer which is SoftMax layer itself.

Below figure describes all the processes occurring inside a typical CNN in a nutshell:-

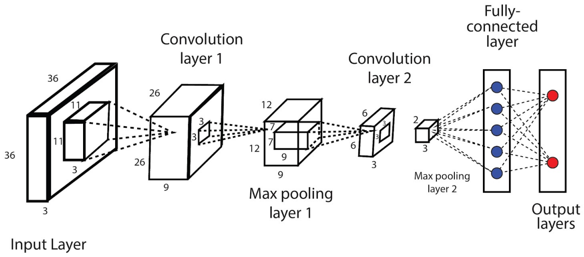


FIG 6: Simple CNN model

Architecture	Output	Param
Conv. Block 01	(62, 62, 32)	896
Conv. Block 02	(29, 29, 64)	18496
Conv. Block 04	(12, 12, 64)	36928
Flatten	(,9216)	0
Dense	(,3)	27651

Table 2: CNN Architecture and trainable parameters

D. HyperParameter Setting

CNN-based network implemented use Tensorflow framework. Network training make use of Adam optimizer with 3 training epochs, with mini batch size of 100.

Table 1: Setting of spectrogram transformation

Factors	Setting
Spectrogram	Gammatone
Window size	2208
Hop size	256
FFT number	4096
Filter banks number	128
Min frequency	10 Hz

E. WebApp Dhvani

WebApp Dhvani is a simple web based interface that takes a sound input (in wav format) and outputs the prediction result into 3 classes Industrial, Commercial and Residential.

There are a bunch of intermediary processes before final prediction actually happens. Firstly, we need to make sure the input audio is in wav format. If the user inputs an audio in some other audio format, conversion is done in the background using a script. It is preferred that the user provides an audio that has a length of 10 seconds, otherwise audio will be clipped. The output will then be processed forward into the gammatone converter and produces a spectrogram image of the corresponding audio. This image is then converted into a normalised array and fed into a previously trained model to produce a one hot encoded matrix of 3x1 dimensions. Corresponding to the index having value of 1, the results are declared accordingly.

F. Project Pipeline:-

Entire process from training to testing can be summarized within the following key points:-

1. Spectrogram image extraction from training and validation folders. Extraction process is completely random and done in batches of 32 each. We made sure that the samples are well shuffled with a seed value set to a constant integer.

2. Conversion of images into a multidimensional numpy array of size 64 x 64 x 3. Note that the original image can have any random size (in our dataset, original images have a dimension of 640 x 480 x 3). Numpy was essential to avail advantage of faster vectorization operation as the underlying foundation of Numpy library is C.

3. Multidimensional array produced is then passed into our CNN model. At the end of the entire process 83,971 trainable parameters were generated for a single batch.

4. Finally, training was performed using backpropagation. 3 epochs were enough to achieve the desired accuracy. Each epoch was divided into 353 steps.

5. Trained model was then saved successfully and finally deployed for prediction.

V. Results and Discussions:-

This report has presented a robust framework to determine land in use using CNN model. As a result, we have achieved great accuracy of 97 % in our manually prepared validation set.

a) Limitations:- From our observation, the change in quality of sound can affect the prediction results. The spectrogram images produce different output when the same audio is captured using two radically different devices. This can be overcome by using similar audio devices with the same audio capturing quality.

Conversion of audio into a spectrogram takes 3-4 seconds. For a real time system, this can create large computational overhead. This could be improved further by optimizing the conversion part.

There are some audio samples which are just ambiguous and cannot belong to either of the classes. A classic example of dog barking cannot be classified until and unless there is some additional background information describing the environment. This can be overcome further by supporting the model with some other sensor such as thermal sensor which can give better estimation about the environment.

Even though Gammatone spectrograms provide a visual representation of audio as close to that perceived by our human cochlea, they are not perfect. Sometimes we need to accommodate the time series considerations to make better predictions about the environment. For such scenarios RNN or a hybrid of RNN and CNN (also known as CRNN) can be utilised.

Audio length clip limit of 10 seconds is essential to produce uniform size spectrograms however such clips might not be always available. Clipping the audio can result in the loss of important information. This is a challenge we would like to overcome in our future goals.

b) Scope :- This project showcases the promising aspect of CNN based prediction in Acoustic Scene Classification. Predicting land use is beneficial for urban planners and environment enthusiasts. to create informed decisions.

Creating a state of the art system that can determine the former successfully and quickly is required to accommodate the needs of researchers.

c) Acknowledgements :- We would like to express our gratitude towards Dr.Vaibhav Kumar, Professor of Artificial Intelligence (AI) for providing us such a wonderful opportunity and guiding us throughout this journey. We would also like to acknowledge URBANSOUND8K DATASET and TAU Urban Acoustic Scenes 2020 Mobile, Development dataset and YouTube channels for helping us with their content which has been used to prepare the dataset.

VI. Conclusions

In this report, we discussed a methodology for predicting land use using acoustic scene classification (ASC). We studied how audio can be converted and approximated to a visual representation called spectrogram and employed Gammatone Spectrogram to achieve our goal. We employed a low complexity deep learning model (DCNN) and evaluated it's performance. We thereby concluded that Gammatone spectrogram are reliable sources of data for most of the ASC tasks. Finally, we were successful in predicting and deploying our model and created an interface that could take an input and produce a prediction output. This makes our model user friendly for people with a non-technical background and to some extent determines its success. We also discussed some of the limitations of our model and future scope of the project. The model needs lots of data to function in a real world scenario hence requires lots of training but given the time constraints and small amount of data, the results were beyond expectations.

References

[1] R.F. Lyon (2017). ”Human and Machine Hearing”, in Cambridge University Press.

[2] L.Ma; D.J.Smith; B.P.Milner. ”Context awareness using environmental noise classification”, in Eighth European Conference on Speech Communication and Technology, 2003.

[3] Asifullah Khan; Anabia Sohail; Umme Zahoora; Aqsa Saeed Qureshi. ”A Survey of the Recent Architectures of Deep Convolutional Neural Networks”, in Artificial Intelligence Review, 2019.

[4] Alexandru Cazan; Radu Varbanescu; Dan Popescu. ”Algorithms and Techniques for Image to Sound Conversion for Helping the Visually Impaired People - Application Proposal”, in 6th EURASIP Conference focused on Speech and Image Processing, Multimedia Communications and Services.

[5] X Zhao; DL Wang. ”Analyzing noise robustness of MFCC and GFCC features in speaker identification.”, IEEE international conference on acoustics, speech and signal processing (ICASSP), Vancouver, Canada, 26–31 May 2013, pp. 7204–7208

[6] B Glasberg; B Moore. ”Derivation of auditory filter shapes from notched-noise data.”, Hearing Res. 47, 103–108 (1990)

[7] Ivars Namatevs. ”Deep Convolutional Neural Networks: Structure, Feature Extraction and Training”, Information Technology and Management Science 20(1), 2017