Scientific Journal of Impact Factor (SJIF): 5.71

e-ISSN (O): 2348-4470
p-ISSN (P): 2348-6406

International Journal of Advance Engineering and Research Development

Volume 6, Issue 04, April -2019

SOUND RECOGNITION OF SPECIES USING TENSOR FLOW

B.Sri Vidya

4th semester M.Tech, Department of Information and technology
GITAM (deemed to be university)
HYDERABAD

Abstract— *Animal sound recognition is the technological advancement in the audio recognition using machine Learning and deep learning. The audio recognition is traditionally focussed on the speech. The main purpose of animal sound recognition is to recognise the emotions of the species as animals and birds are tending to change their activities as well as their habitats due to the adverse effects on the environment or due to other natural or man-made calamities. The best way to monitor the species is by audio recognition. We train the machines by pre-recorded audio files. And these audio files should have no background noise or echoes to avoid overlapping.*

We here use convolutional neural networks using tensor flow and train the audio datasets of the species. The convolutional neural networks work on the frequency cepstral coefficients that are extracted from the audio datasets. The manuscript serves as a technical paper showing how the model works to achieve the desired result.

Keywords— *Audio recognition, Convolutional Neural Networks, Deep Learning.*

I. INTRODUCTION

As a part of the ecosystem, we humans need to need to make sure the animals and birds living around us as well as in the forest are well protected for various reasons. 80% of biodiversity on land lives in the forests which signifies the importance it has in maintaining the ecological balance. It is also important to understand how humans are effecting the animals and their ecosystem in the forest. This method not only would be helpful in finding how the animals are effected by our footprint but also can help us warn about a lot of natural calamities beforehand.

As our model includes audio datasets with and without background noise, we can even take background noise into consideration and execute various methods which will help in saving the environment. We can even record various background noise of forest fires, cutting down of trees with equipment and many more which are threat to the forests and the environment.

Along with the research of Automatic Speech Recognition (ASR) and Music Information Retrieval (MIR), the research field of Automatic Bird Identification (ABI) has received special attention only in the past decades (Tivarekar, 2017). Bird and Animal audio detection is one of the most intensive researches given by Detection and Classification of Acoustic Scenes and Events (DCASE 2018), since birds and animals are more easily detectable through the audio modality rather than vision. Traditional method used to recognize animal and bird sounds are from the generated spectrograms and continuous monitoring of those spectrograms for recognition of the species is a tedious task.

In this report, we will find an approach which is used to identify and detect birds and animal sounds with the help of pre-recorded datasets.
Two layer convolution networks are used to detect human speech commands in tensor flow. We will use the similar model along with animal datasets instead of human. We will then convert all the .mp3 files to .wav files. All these .wav files are modulated in such a way that every audio file is having a similar frequency of 16 000 Hz and bit rate of 16-bit. These are then trained by machines and are recognized with the help of three main factors such as rate, cross entropy and accuracy. The clearer the dataset is, the higher accuracy will be.

To get a clear audio file without any background noise, we need to have pre-recorded sounds of all the possible background noise and make sure we cancel these noises in the considered dataset and after many approaches.

Let's have a look at how audio recognition in species is important:

Firstly, the audio frequencies of birds carry a lot of information which can be either about climate change or about their habitats. With the help of various frequency ranges, we can help the species located at that habitat and save them if there are any problems. One other reason for change in frequencies might be due to lack of food.

Secondly, Continuous monitoring of species is a tedious task. It requires a lot of labour along with a lot of advanced equipment which causes organizations to spend a lot of money, time and labour. With pre-recorded audio datasets, we can know if the species are in danger or not. A lot of money as well as time will be saved.

Finally, we can protect the forests from forest fires as we will have pre-recorded audio dataset of forest fire. If this audio is recognized, then we can send rescue team on spot. We can even protect forests from deforestation. If there is any other sound, then other variable will be triggered and can send signals to the receiver. This can help forests along with the Species, providing better and safe habitat for them to co-exist in

II. LITERATURE REVIEW

In this section, instead of covering broadly how machine learning is applied nowadays (e.g., internet of things (Zeng, E-AUA: An Efficient Anonymous User Authentication Protocol for Mobile IoT. IEEE Internet of Things Journal., 2018) (Zheng, 2014), social networks (Wang, 2018), activity recognition (Bhandari, 2017) (Pan, 2018), recommendation (Fu)), we will detailed information about the Audio Recognition done with the help of Tensor Flow.

"According to Sophia, Tensor Flow is used to implement complex DNN structures without getting complex mathematical details, and availability of large datasets. The machine learning model used in this paper is CNN which consists of three hidden layers." ([1] Thakare). This algorithm predicts the bird sounds and gives output as 1 if found the audio convolutional neural network is finely-tuned and are trained from the scratch." ([2] Koitka)

Dorota Kamiska and Artur Gmerek presented fully automated algorithm ([3] kaminska, (2012)). SOM and k-NN classifiers, have been chosen and compared in their paper for sounds of few species downloaded from various web sources (Disjoint sets with 70% - for preparing, 30% - for testing). The order precision for various highlights demonstrated that spectral features are the best for Automatic Species Recognition task. Their best outcomes had mean Classification precision of 69.94% with k-NN classifier and 52.92% with SOM classifier.

Chang-Hsing Lee et al. used frequency information to extract the syllables exactly ([4] Lee, (2006)). Averaged MFCCs in a syllable were used to identify species from their sounds. Experiments concluded that AMFCC greatly outperforms HMM and ALPC in training and testing. The average classification accuracy was up to 96.8% and 98.1% for frog calls and cricket calls, respectively.

Panu Somervuo, Aki Harma, and Seppo Fagerlund segmented a recording into individual syllables using a time-domain algorithm and segmented each region using three models such as Sinusoidal Model, Mel-Cestrum Model, and Descriptive Parameters. Dynamic time warping (DTW) algorithm was used for comparing variable length sequences ([5] Somervu, (2006). Gaussian mixtures were used for modelling probability density functions in pattern recognition. The average recognition accuracy for single syllable was only around 40% to 50% depending on the method. The recognition results improved significantly in song-based recognition.

Iosif Mporas et al. evaluated the appropriateness of bird species recognition task with the help of real-field audio recordings of seven bird species, which are common for the Hymettus Mountain in Attica, Greece (Mporas, 2012). Two temporal and sixteen spectral audio descriptors comp ([6] Mporas)uted using the open SMILE acoustic parameterization tool were used. For high SNR the boosting algorithm outperformed all the rest classification algorithms, while for low SNR the bagging meta- classifier offered slightly better performance than the boosting algorithm with maximum classification accuracy of 92.89%.

III. KEY RESEARCH AREAS

A. Extracting Frequency Domain Representation

Recognizing animals and birds sound is a tough task for a machine than compared to human brain. To train, we used two different datasets namely animals and birds. Both datasets contain short sound clips of various species. There is only a single event present in each sound file thus preventing overlapping.

B. Using Animal Datasets

In this dataset, we took in two different animals naming the dataset folder as Bark for dog, Miaow for cat. We've considered two animals so that we can work faster on a small dataset. The average duration of each audio dataset file is 1.0 s.

C. Using Bird Datasets

In this dataset, we took in two different birds naming the dataset folder as Pigeon and Peacock. The average duration for each audio dataset file is 1.0 s.

D. Bandpass filtering
Most of the training datasets are under 500 Hz and above 18 000 Hz this causes a lot of problem while testing on different datasets and training outcomes will be inaccurate. To maintain the integrity among all the datasets, we need to increase and decrease the Hz of datasets and make every dataset equal to 16 000 Hz.

E. Noise Filtering
Here we will reduce the background noise in the datasets by assuming few background noises. We take waterfall, heavy wind and many other scenarios into consideration and make sure we will reduce these background noises if found in the audio datasets so that we can train the machine with only animal or bird sound.

F. Silent Region Removal
Few audio sources are found where we can hear animals and birds sound in an echo. This echoing effect will increase or decrease the frequency and modulation of voice and it will cause a lot of trouble in detecting the species. We use this technique to reduce such sounds.

IV. METHODOLOGY

A. First Stage
Initially we found data sets containing human voices on Tensor Flow. The deep neural networks of Tensor Flow are considerably efficient as their processing speed is very high which leads to a very low processing time. We now understand the algorithm used on Tensor Flow to recognize and categorize human voice commands. The python file used to train data sets uses CNN. CNN considers three factors which are Rate, Accuracy, and Cross Entropy as parameters. It makes modulations to these parameters and checks at every modified or desired steps where each step is a different value assigned to Rate and Cross Entropy to form a confusion matrix. Every confusion matrix has different combination of Rate and Cross Entropy with which it is trying to determine the best way to achieve highest accuracy possible. After 18000 steps, it compares all the confusion matrices formed and deciphers the best combination of Rate and Cross Entropy where the Accuracy is at its highest.
Before executing following steps, we need to make sure that we've installed Python correctly and should make sure if the machine on which we are training can support GPU functionality or not. Generally, machines which support GPU functionality will have a capability to train faster than the machines which only support CPU functionality.

Step-1: We need to download the python files from the Tensor Flow website.
Step-2: Now we need to open command prompt and navigate to the program path and execute train code.
Step-3: As soon as we execute the train.py file, the Python program will download human command datasets from cloud repositories.

```
I0730 16:57:38.073667   55030 train.py:243] Confusion Matrix:
 [[258   0   0   0   0   0   0   0   0   0   0   0]
 [  7   6  26  94   7  49   1  15  40   2   0  11]
 [ 10   1 107  80  13  22   0  13  10   1   0   4]
 [  1   3  16 163   6  48   0   5  10   1   0  17]
 [ 15   1  17 114  55  13   0   9  22   5   0   9]
 [  1   1   6  97   3  87   1  12  46   0   0  10]
 [  8   6  86  84  13  24   1   9   9   1   0   6]
 [  9   3  32 112   9  26   1  36  19   0   0   9]
 [  8   2  12  94   9  52   0   6  72   0   0   2]
 [ 16   1  39  74  29  42   0   6  37   9   0   3]
 [ 15   6  17  71  50  37   0   6  32   2   1   9]
 [ 11   1   6 151   5  42   0   8  16   0   0  20]]
```

Figure 1: Confusion matrix

This is known as Confusion Matrix. Each column represents a set of samples that were predicted to be each label, so first two rows belong to background noise and third row belongs to other sound. The remaining rows belong to individual speech command datasets ("yes", "forward", "backward" etc.).

Each row represents clips by their correct, ground truth labels. This matrix can be more useful than just a single accuracy score because it gives a good summary of what mistakes the network is making.

A perfect model would produce a confusion matrix where all the entries were zero apart from a diagonal line through the centre. Spotting deviations from that pattern can help you figure out how the model is most easily confused, and once you've identified the problems you can address them by adding more data or cleaning up categories.

Step-5: As soon as 18 000 steps get executed, we will get final matrix along with the maximum possible detection accuracy of the provided datasets.
Step-6: We've now successfully trained machine and now we need to check if the machine is recognizing commands or not

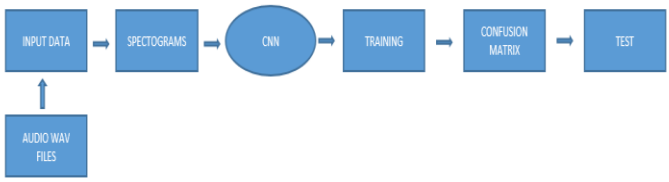

Figure 2: METHOD

V. METHOD AND ALGORITHM USED.

A. Algorithm

Convolutional Neural Networks are very similar to ordinary Neural Networks. They are made up of neurons that have learnable weights and biases. Each neuron receives some inputs, performs a dot product and optionally follows it with a non-linearity. The whole network still expresses a single differentiable score function: from the raw image pixels on one end to class scores at the other. And they still have a loss function (e.g. SVM/Softmax) on the last (fully-connected) layer.

Neural Networks receive an input (a single vector), and transform it through a series of hidden layers. Each hidden layer is made up of a set of neurons, where each neuron is fully connected to all neurons in the previous layer, and where neurons in a single layer function completely independently and do not share any connections. The last fully-connected layer is called the "output layer" and in classification settings it represents the class scores.

3D volumes of neurons: Convolutional Neural Networks take advantage of the fact that the input consists of images and they constrain the architecture in a more sensible way. In particular, unlike a regular Neural Network, the layers of a ConvNet have neurons arranged in 3 dimensions: **width, height, depth**. (Note that the word depth here refers to the third dimension of an activation volume, not to the depth of a full Neural Network, which can refer to the total number of layers in a network.)

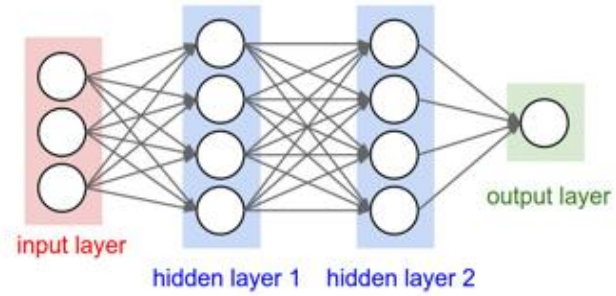

Figure 3: Architecture of CNN

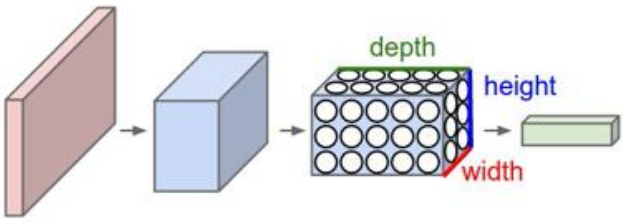

Figure 4: Representation of filter

B. Feature Extraction

This stage provides actual representation of a waveform created by speech. This feature extraction method will minimize the loss of information by providing accurate distributional assumptions made by acoustic model. For all state output distributions, we used Gaussian distributions. This method is best suitable for convolutional networks.

All the encoding schemes are based on mel-frequency cepstral coefficients (MFCCs). These cepstral coefficients are considered to smooth the audio and helps in reducing the background noise. The smoothing of an FFT is done around 20 Frequency bins. The non-linear frequency scale used here is known as mel scale and this has a similar response to that of human ear.

We need to download ideal datasets. Ideal datasets are those which are pre-recorded without any disturbance in spectral waves and there is no disturbance in background. The accuracy of the detection along with the displayed confusion matrix depends upon the datasets. The clearer the dataset, the higher is the accuracy we will achieve.

We need to make sure that all the audio files in the datasets are of similar frequency along with the bit rates. If one file is having 8-bit rate and if other file is having 16-bit rate, then we will have mel-frequency error where the files frequency will be mismatched. So, we need to make sure that all the files are

1. Set to 16-bit rate.
2. All the files are set as mono audio.
3. All the files should have a frequency range of 16000Hz
4. The duration of each audio file should be 1 second.

The reason behind mono audio is that many audio files will have a similar bit rate and frequency. If their range is differentiating, then there will be a problem in creating spectrograms which are the main reason in detecting audio. If the files are having mono audio, then we need to leave as it is. If the files are having bi-audio, then we need to make sure that average value range is considered.

To do this process, we need to convert all the .mp3 files to .wav files and during the converting process, we need to use some pre-defined tools which help in aligning all the datasets equally.

In our training program, we are considering our frequency range to be 16 000Hz. We need to make sure that we convert the files to that range.

We can even change the parameters if required. The reason behind considering audio files to be 1 second is due to the processing time and about the spectrograms. The accuracy of detection may vary depending upon change in time duration as well as the bit rate.

VI. TRAINING

The parameters which are taken into consideration are:

1. Resolution – 16 Bit
2. Sampling Rate – 16 000 Hz
3. Audio Channel – mono
4. Audio Trim – 1 Second

We work on 6 datasets namely,

1. Silence
2. Unknown
3. Bark
4. Miaow
5. Pigeon
6. Peacock

Depending upon the dataset as well as the sampling rate, we can see that different confusion matrices will be formed. After successful 9000 steps, we get validation accuracy to be 100%. And our final test accuracy to be 55.6%.

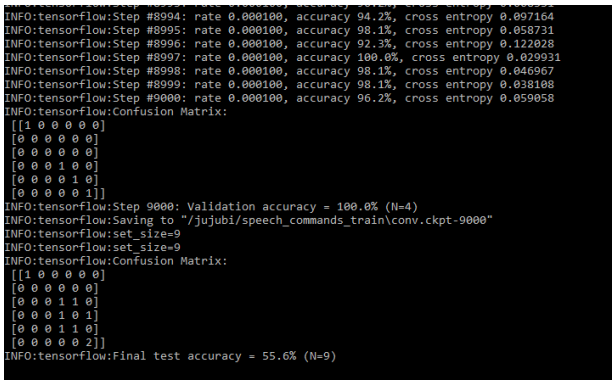

Figure 5: Regular model confusion matrix

The confusion matrix is an ideal one when there is only 1's and 0's in it. As we can see, our confusion matrix is almost ideal and if we test on an audio then we will get its probability of recognition.

A.A Testing on different audio datasets

1. Dog

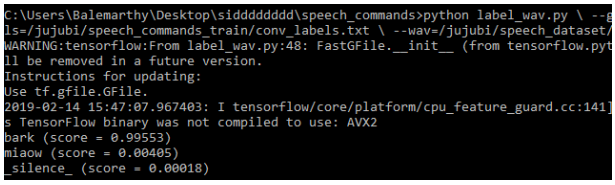

Figure 6: Prediction of Dog

As we can see, we got a probability of 0.99 in detecting dog audio. We can consider that this file is ideal for recognition of dog species.

2. Cat

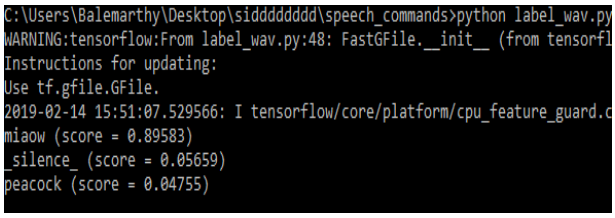

Figure 7: Prediction of Cat

As we can see, we got a probability of 0.89 in detecting cat audio. We can consider that this file is almost ideal for recognition of cat species.

3. Pigeon

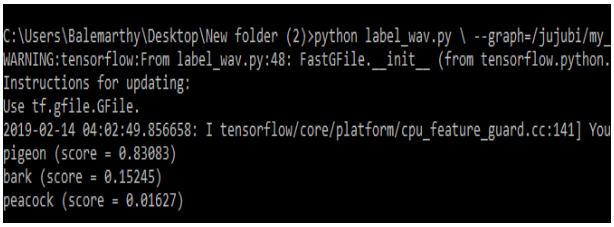

Figure 8: Prediction of Pigeon

We got a probability of 0.83 in detecting Pigeon audio. We can consider that this file is almost ideal for recognition of Pigeon species.

4. Peacock

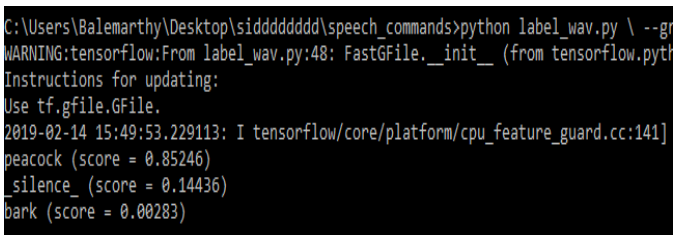

Figure 9: Prediction of Peacock

We got a probability of 0.85 in detecting Peacock audio. We can consider that this file is ideal for recognition of Peacock species.

By considering the probability of recognition, we can thus conclude that the audio datasets are ideal. The probability in detection is almost 1. To increase the train and test accuracy, we trained the datasets by adding an extra layer.

VII. TRAINING WITH AN ENHANCED MODEL

We don't change the parameters of the audio datasets but we just add the layers in the convolutional model. After training the datasets with 9000 steps with the new model we get the following result containing the corresponding confusion matrix and the respective validation and test accuracy.

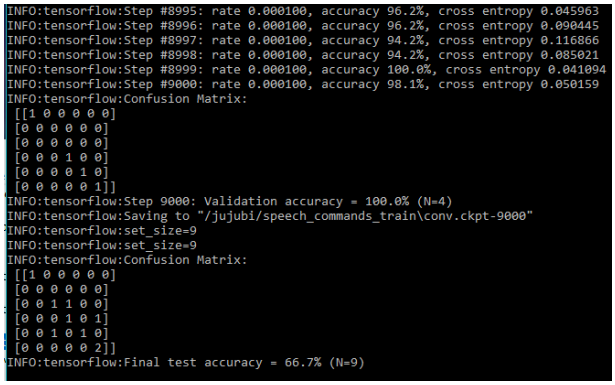

Figure 10: Enhanced model test accuracy

After successful 9000 steps, we get validation accuracy to be 100%. And our final test accuracy to be 66.7%.
We can see a significant development in the test accuracy on the same dataset.

B.A Testing on different audio datasets

1. Dog

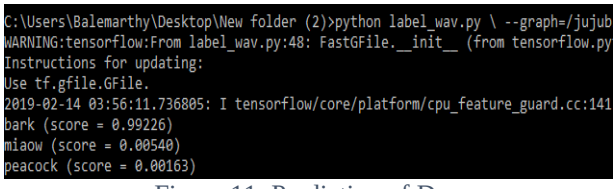

Figure 11: Prediction of Dog

As we can see, we got a probability of 0.99 in detecting dog audio. We can consider that this file is ideal for recognition of dog species.

2. Cat

Figure 12: Prediction of Cat

As we can see, we got a probability of 0.75 in detecting cat audio. We can consider that this file is almost ideal for recognition of cat species.

3. Pigeon

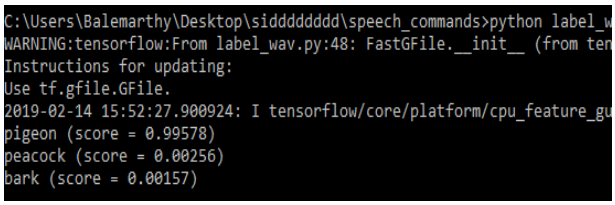

Figure 13: Prediction of Pigeon

We got a probability of 0.99 in detecting Pigeon audio. We can consider that this file is ideal for recognition of Pigeon species.

4. Peacock

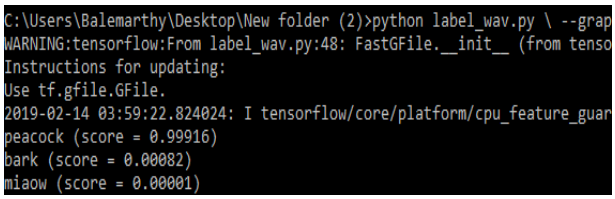

Figure 14: Prediction of Peacock

We got a probability of 0.99 in detecting Peacock audio. We can consider that this file is ideal for recognition of Peacock species.

We can even change the parameters of the datasets and use the model and train them to get required results. Different parameters can be change in the frequency by 8000Hz, changing the sampling channel from mono to stereo and also the sampling rate.

VIII. DISCUSSION OF RESULTS

In order to prove our method, we have chosen 4 different species with 20 audio datasets each. All of these audio datasets are pre-recorded wherein few of these audio datasets have background noise and few don't. The overall accuracy is decided by three parameters such as frequency, audio rate, sampling channel. All of these are recorded with a time length of 100 milliseconds.

We will now test which method along with which model is best suitable to train data. We have used a model with two layer CNN as well as a 3 layer CNN.
We here compare the accuracy of the both the models on the same datasets.

Model	Validation accuracy	Test accuracy
original	100%	55.6%
enhanced	100%	66.7%

Table 1: comparison of models

We need to now compare the test cases while testing on a particular audio data file.

Model	Dog	Cat	Pigeon	peacock
original	0.99	0.89	0.83	0.85
enhanced	0.99	0.75	0.99	0.99

Table 2: comparison of test results

By considering the above displayed results generated by test cases, we can clearly deduce that adding more layers is the best approach to recognize the audio datasets and by considering this approach we get the total accuracy as 66.7%.
In the real time, we might have thousands of data sets and audio files. In order to predict the required audio we need more layered network. This can be the ideal approach.

IX. PROBLEM ANALYSIS

This recognition ability is tested on few audio datasets. All the accuracies have only been recorded for limited set of species. Let us assume two species/animals which are fox and dogs. Both species have same sound while howling. There will be the slightest difference in some ranging syllable which cannot be detected by humans. This method might be able to generate spectrograms of both species howling. But it might also not be able to spot the difference. To reduce this kind of problems, we can either create an audio recognition for species which are domestic separately, or we can even increase the time duration of both audios and then find difference while lowering their voices.

Secondly, if we consider Parrot which is a bird species, it has the ability to sound like humans and other objects. This will confuse the machine to recognize if it is human's voice or the audio of parrot. It can even reciprocate background noise of various audio datasets wherein we will get the output recognized audio as "other" or "unknown". To reduce this type of problems, we can have an approach where we can see the pitch variation difference in each syllable and find out if it is parrot or human or any other destruction caused to the forest.

Finally, all the audio datasets in this method are predefined to detect them. If there is any case where we can update the audio datasets with lot of possibility audio produced by respective species, then this will help in recognizing the species without any loss in validation accuracy as well as in overall accuracy. To detect the species perfectly without any loss we can increase the sound in the audio datasets along with increasing the time duration of the every individual audio of the species.

X. CONCLUSION

In this report, we have given an overview about the audio recognition of species and how they are helpful in real lives. We also provided the usage and working of commonly used Convolutional Neural Network. We have provided different stages of execution along with the detail approach which helped in recognizing audio. Our method still needs improvement such as decreasing or increasing the bit rate along with the increase in audio duration. This can help in improving the overall accuracy even though this method is successfully recognizing the audio without any trouble. So, in the end our model can recognize the audio of species without any errors. Our work can help resolve interesting research challenges saving species from being endangered

XI. REFERENCES

[1] Thakare, D. (. (n.d.). Extracting Frequency Domain Representation.
[2] Koitka, S. (. (n.d.). Recognizing Bird Species in Audio Files Using Transfer Learning.
[3] kaminska, D. .. ((2012)). Automatic identification of bird species: A comparison between KNN and SOM classifiers.
[4] Lee, H. .. ((2006)). Automatic recognition of animal vocalizations using aver- aged MFCC & linear discriminant analysis.
[5] Somervu, P. ((2006). Parametric representations of Bird Sounds for Automatic Species Recognition.
[6] Mporas, I. (. (n.d.). Automated acoustic classification of bird species from real-field recordings.

