Deep-Learning Based Sound Recognition To Classify Sounds (Using Convolutional Neural Networks)

Pavan kamal Pullabhotla[1], Rohit Chaudhari[2], Shreya Babulkar[3], Vini Dubey[4], Prof. Rajesh Tak[5]
[1, 2, 3, 4, 5] Dhole Patil College of Engineering, Wagholi, Pune

Abstract- *Since the 1950s, the early days of artificial intelligence, computer scientists have been trying to build computers that can make sense of visual data. In the ensuing decades, the field, which has become known as computer vision, saw incremental advances. Convolutional neural network (CNN), a specialized type of artificial neural network that roughly mimics the human vision system. In recent years, CNNs have become pivotal to many computer vision applications. Convolutional neural networks are composed of multiple layers of artificial neurons. Artificial neurons, a rough imitation of their biological counterparts, are mathematical functions that calculate the weighted sum of multiple inputs and output an activation value.*

I. INTRODUCTION

In the 1950s and 1960s **David Hubel** and **Torsten Wiesel** conducted experiments on the brain of mammals and suggested a model for how mammals perceive the world visually. Our research on vision has been going on since then. Out of such research was born a very powerful algorithm known as the **Convolutional Neural Network (CNN).**

Using CNN to classify sounds one has to convert sound frequencies into images. CNN takes images as input, converting the sound files to mel-spectrograms can be a valid solution in this case. Running a CNN needs data, so we have borrowed a few sound samples from **UrbanSounds8k** by J. Salamon and J. P. Bello. This CSV file consists of 8732 files (<=4s) classified into 10 different classes i.e. air_conditioner, car_horn, children_playing, dog_bark, drilling, enginge_idling, gun_shot, jackhammer, siren, and street_music.

Convolutional neural networks have a high model capacity, and are particularly dependent on the availability of large quantities of training data in order to learn a nonlinear function from input to output that generalizes well and yields high classification accuracy. **Data Augmentation** is a valid solution to get data in the same classes. Using a library named Librosa one can export data from pre classified data given in Urbansounds8k file and export them by reshaping the data by changing hyper parameters of the input data in this case which are random combinations of time shifting, pitch shifting and time stretching.

II. METHOD

The Convolution Neural Network architecture proposed in this paper consists of 3 convolutional layers interleaved with 2 pooling operations, followed by 2 fully connected (dense) layers. The input to our neural network consists of TF-patches i.e. time frequency patches extracted from log scaled Mel-spectrogram representation of the audio signal. **Mel-spectrograms** can be generated by using a library named **Librosa.** Librosa's feature to extract mel-spectrograms was used to generate log scaled mel-spectrograms with 128 bands.

Given input to the Neural Network was restricted to 3 seconds (128 frames) i.e. . $X \in R^{128\times128}$

Given input X, the network is trained to learn the parameters Θ of a composite nonlinear function $F(\cdot|\Theta)$ which maps X to the output (prediction) Z:

$$Z = F(X|\Theta) = fL(\cdot\cdot\cdot f2(f1(X|\theta1)|\theta2)|\thetaL),$$

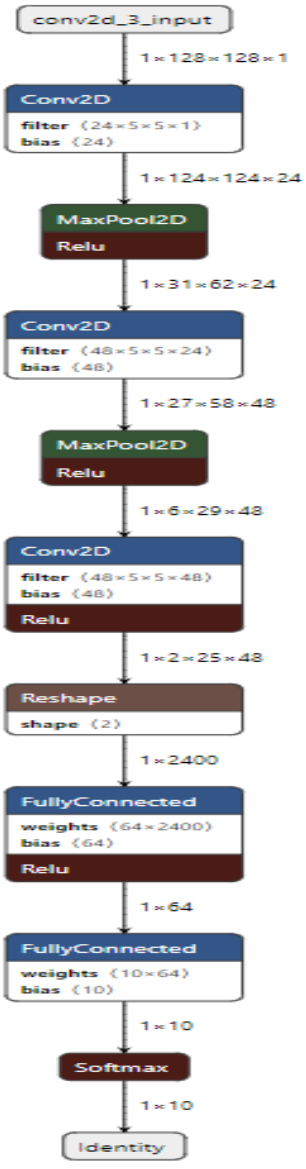
where each function $f`(\cdot|\theta`)$ acts as a layer in the network, with L = 5 layers in our proposed architecture. The first three layers, ` ∈ {1, 2, 3}, are convolutional, expressed as:

$$Z` = f`(X`|\theta`) = h(W * X` + b), \theta l = [W, b]$$

The proposed CNN architecture is parameterized as follows:

- L1: 24 filters with a receptive field of (5,5), i.e., W has the shape (24,1,5,5). This is followed by (4,2) strided max pooling over the last two dimensions (time and frequency respectively) and a rectified linear unit (ReLU) activation function h(x) = max(x, 0).
- L2: 48 filters with a receptive field of (5,5), i.e., W has the shape (48, 24, 5, 5). Like L1, this is followed by (4,2) strided max-pooling and a ReLU activation function.

- L3: 48 filters with a receptive field of (5,5), i.e., W has the shape (48, 48, 5, 5). This is followed by a ReLU activation function (no pooling).
- L4: 64 hidden units, i.e., W has the shape (2400, 64), followed by a ReLU activation function.L5: 10 output units, i.e W has the shape (64,10), followed by a softmax activation function using Adam optimizer. Adam combines the best properties of the AdaGrad and RMSProp algorithms to provide an optimization algorithm that can handle sparse gradients on noisy problems.
- Categorical cross-entropy is used as a loss function since we are doing multi-class classification tasks.



Primal Attempts :

- Epochs = **15**, Batch_size = **64** : Test accuracy = **73.4%**, Test Loss = **97.2%**
- Epochs = **20**, Batch_size = **64** : Test accuracy = **72.1%**, Test Loss = **93.8%**
- Epochs = **20**, Batch_size = **32** : Test accuracy = **77.7%**, Test Loss = **75.7%**
- Epochs = **25**, Batch_size = **32** : Test accuracy = **83.2%**, Test Loss = **51.0%**

Data Augmentation :

Using the data provided by Urbansounds8K we could pull out a test accuracy of **84.5%**. As CNN's need large quantities of training data we have used Librosa to extract augmented sound files using recursive functions and exported approximately **29000** sound files.

Note that for each augmentation it is important that we choose the hyperparameters. The hyperparameters and resulting augmentations are described below:

- **Time Stretching (TS):** Using time stretch one can slow down or speed up the given audio. While we use time stretch to change the speed we keep the audio's pitch unchanged. Each sample was time shifted by **1.07**.
- **Pitch Shift (PS1):** Using Pitch Shifting we can change the pitch of sound files keeping its duration unchanged. Each sample was shifted by **2**.
- **Pitch Shift (PS2):** We have created a second augmentation using pitch shift and each sample was shifted by **2.5**.

Post augmentation the data we could extract **29835** sound files. Appending all of data to D we could use it again to the model and run it to give a better test accuracy of **90%**

Final Attempts :

- Epochs = **35**, Batch_size = **32** : Test accuracy = **90.6%**, Test Loss = **31.7%**
- Epochs = **50**, Batch_size = **64** : Test accuracy = **93.7%**, Test Loss = **19.5%**
- Epochs = **50**, Batch_size = **64** : Test accuracy = **92.8%**, Test Loss = **22.7%**

With the augmented data best test accuracy and test loss was found when epochs = 50 and batch_size = 64.

The Classification phase comprises a deep neural network based on Convolutional neural networks which are well suited to classification problems. The convolution in the

convolution stage of a single convolution is achieved through kernels that are convolved with the RGB image. In case of multiple kernels all feature maps obtained from distinct kernels are stacked to get the final output of that layer.

To introduce a non-linearity to the linear convolution operation, A non-linear Activation function layer is introduced, for example, ReLU. Now the Pooling stage of the convolution layer replaces the output of a node at certain locations with a summary statistic of nearby locations and pooling can be of different types : Max, Average, Sum, etc. The max pooling reports the maximum output within a rectangular neighbourhood. Pooling helps to make the output approximately invariant to small translation i.e. it reduces the dimensionality of the spectrogram.

III. CONCLUSION

In this paper we proposed a deep convolutional neural network architecture which, in combination with a set of audio data augmentations, produces usable results for urban sound classification. We showed that the improved performance stems from the combination of a deep, high-capacity model and an augmented training set: this combination significantly improves over the proposed CNN without augmentation. Finally, we examined the influence of each augmentation on the model's classification accuracy. We observed that the performance of the model for each sound class is influenced differently by each augmentation set, suggesting that the performance of the model could be improved further by applying class-conditional data augmentation.

REFERENCES

[1] Justin Salamon and Juan Pablo Bello "Deep Convolutional Neural Networks and Data Augmentation for Environmental Sound Classification" *IEEE SIGNAL PROCESSING LETTERS, ACCEPTED NOVEMBER 2016*

[2] Omkar Chavan, Nikhil Kharade, Amol Chaudhari, Nikhil Bhalke, Prof. Pravin Nimbalkar, "Machine Learning and Noise Reduction Techniques for Music Genre Classification" *Volume: 06 Issue: 12 | Dec 2019, p-ISSN: 2395-0072, e-ISSN: 2395-0056.*

[3] Ying-Hui Lai, Yu Tsao, Xugang Lu, Fei Chen, Yu-Ting Su, Kuang-Chao Chen, Yu-Hsuan Chen, Li-Ching Chen, Lieber Po-Hung Li, Chin-Hui Lee, "Deep Learning–Based Noise Reduction Approach to Improve Speech Intelligibility for Cochlear Implant Recipients." *Article in Ear and Hearing · January 2018, Lai et al. / EAR & HEARING, VOL. XX, NO. XX, 00–00.*

[4] A. Zorzo, W. D'A. Fonseca , E. Brandão , P. H. Mareze, "Design and analysis of a digital active noise control system for headphones implemented in an Arduino compatible microcontroller." *ArtigoSIIM-SPS2017 November 2017.*

[5] Abhishek Manoj Sharma, "Speaker Recognition Using Machine Learning Techniques", *San Jose State University " (2019). Master's Projects. 685. DOI: https://doi.org/10.31979/etd.fhhr-49pm.*

[6] Krishna A/L Ravichandran, "Active Noise Reduction using LMS and FxLMS Algorithms" Krishna *A/L Ravinchandra et al 2019 J. Phys.: Conf. Ser. 1228 012064.*