Sādhanā (2020)45:210
https://doi.org/10.1007/s12046-020-01442-x

Deep convolutional network for urbansound classification

N KARTHIKA* and B JANET

Department of Computer Applications, National Institute of Technology, Tiruchirappalli 620015, India
e-mail: bharathikarthika@gmail.com; janet@nitt.edu

MS received 3 August 2019; revised 21 May 2020; accepted 23 June 2020

Abstract. The efficiency of Convolutional Neural Networks in classifying terse audio snippets of UrbanSounds is evaluated. A deep neural model contains two convolutional layers coupled with Maxpooling plus three fully interconnected (dense) layers. The deep neural model is being trained upon low level description of various urban sound clips with deltas. The efficiency of the neural network is examined on urban recordings and compared with different contemporary approaches. The model obtained 76% validation accuracy that is better than other conventional models which relied only on Mel Frequency Cepstral Coefficients.

Keywords. Convolutional neural networks; MaxPooling; rectified linear units (ReLU); UrbanSounds; multi-class classification.

1. Introduction

Detection of everyday ordinary sounds has countless applications such as audio surveillance [1], security monitoring in room [2] and public transport [3], autonomous vehicle [4], detection of intruders in wildlife [5], medical telemonitoring [6] i.e., examining elderly people and observing noise pollution in cities [7, 8]. There are masses of research about sound classification in many fields like bio-acoustics, speech, song and music, but an attempt to investigate the urban sound source environment is exceptionally uncommon [9], because the urban sound sources have a variety of heterogeneous aurals that occur from city acoustic atmosphere, the patterns too differ extensively [10]. The existing approaches usually focus on categorizing the auditory scene type like a park, street [11–14] in opposition to the recognition of particular acoustic sources in those scenery like an engine idling, car-horn or a bird tweet. The latter requires greater effort due to the existence of multiple sources with a variety of mechanisms to produce sound. Further, these can be concealed by noise or some are fairly noise like sounds themselves, for example engine sounds and air conditioners. Most of the past work on urban sound source classification is built on classical [15], hand-crafted features [16, 17] which proved to be over-sensitive to urban background noise environments [18]. Deep convolutional neural networks are absolutely appropriate to classify the urban sound [19].

They are efficient to capture the energy modulation templates over time and frequency when supplied into spectrogram related inputs. It is an important quality to differentiate between different sounds such as gunshot and siren [15].

The Convolutional Neural Network (CNN) is capable to learn fruitfully through making use of convolutional filters (kernels) with a modest receptive field and later recognize spectro-temporal patterns which might be representative of diverse distinctive sound classes, although some portion of aural source is hidden (in time/frequency) by means of additional sources including noise wherein conventional audio features like Mel Frequency Cepstral Coefficients (MFCC) may additionally fail.

Nevertheless, the utilization of CNNs in the urban sound category has been bound to date. The feasible reason for the bounded exposure of CNNs and the hardship to develop on easier models is the relatively insufficient labeled source data. Although certain positive new datasets [8, 20, 21] have been launched in modern-day years, however, they are significantly smaller in size compared to dataset at hand for research, as an example image classification [22]. So, some of the augmentation approaches were used to get rid of scarcity of labeled data, but they proved not to be sufficiently good enough when considering about the Urbansound8K aural data source given the extensive hike in time to train the model to generate and insignificant effect on accuracy of the model [23]. Many researchers have been attracted by aural source classification. They applied different methodologies to classify. [8, 24] applied Support Vector Machine (SVM), [9, 20] used random forest classifier and [25] classified with multilayer perceptron. Lately, [19, 23] applied convolutional neural network and proved that CNN outperforms the conventional approaches, but their model architecture achieved lesser accuracy than JK+CNN model used in this paper.

*For correspondence

Published online: 28 August 2020

2. Convolutional neural networks

Basically, the multi-layer perceptron architecture model's enlargement is well referred to as the Convolutional Neural Network (CNN) model. Still, their architectural distinctness/inequalities have considerable impact. Convolution is also mathematically expressed by an asterisk * symbol. If we do have an input data defined as X as well as a filter defined as f, then representation might be:

$$\mathrm{Z} = \mathrm{X} * \mathrm{f}$$

2.1 *Layers*

An exemplary CNN comprises of some of a distinctive, well organized layers stacked as a group in the deep structure like an information feed layer i.e., input layer, a bundle of convo- lutional layers along with pooling layers (there are a variety of ways to combine those layers), a finite number of completely linked hidden layers as a consequence a loss (output) layer. The primary difference in comparison to the multilayer perceptron, falls in these blend of convolution along with pooling operations which is shown in figure 1.

A convolutional layer proposes a specific manner of formulating/systematizing hidden units which intend to take benefit of the neighborhood arrangement of input data, i.e., prevailing in the mainstream of two-dimensional input records (no longer simply confined to images). Each hidden unit, rather than being linked to entire input cells passing from the preceding network region, is bound to processing best a very small part of the entire input space termed as receptive fields.

The weights belong to any such hidden unit forms a convolution filter (kernel) that is implemented to the complete input space, ended as feature map. In this way, a particular set of determined weights may be reapplied for the entire input data space. This is established on the speculation that locally appropriate feature shall also be useful in some other region of the source data space which is a system not only enormously diminishes the range of parameters to analyze, however complements robustness against translational fluctuation of the information data records. A classical convolutional layer has a lot of kernels, i.e., filters (feature maps).

In addition, a reduction in dimensionality may be attained by pooling layers, that combine the feature map cells which are close to each other. The ultimate pooling operations accomplished are max (topper takes all) or average quantity of the data cells input. This type of down sampling supplementary complements invariance to further adaptation.

2.2 *Rectified linear units (ReLU)*

Logistic sigmoid and hyperbolic tangent were maximum broadly nonlinear activation functions in multilayer perceptron set-up. In recent days, deep architecture has undoubtedly replaced them with another sort of answers. One of the powerful alternatives is Rectified Linear Units, that employs subsequent activation function.

$$f(x) = max(0, x)$$

ReLU has many significant advantages over other activation functions. They are

- Computational efficiency
- Sparsity [26]
- Reduced likelihood of vanishing gradient

Still, it has disadvantages like dead region, because it is based on the kingdom of the random weight utilization, many of the units are prematurely dropped in the dead region area resulting in zero gradient. Due to this, substitutes are given with a nonzero slope like Leaky ReLU [27] and its fruitfulness is proved [28].

2.3 *Dropout learning*

Deep neural models are liable to over-fitting. Still in CNNs, in which the number of parameters is diminished via weight sharing, most of the times, the number of expected values is distinctly greater than the whole number of cases for training by the order of the scale. This could eventually lead out-of-sample.

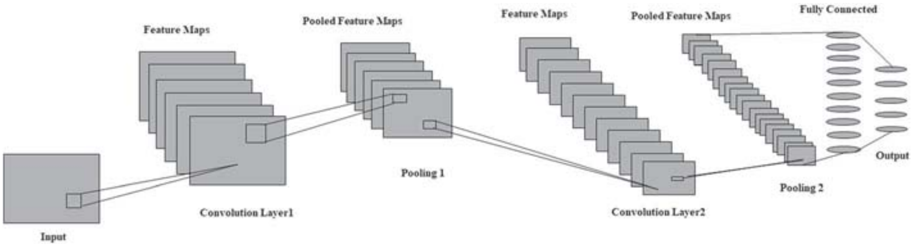


Figure 1. Convolutional Neural Networks.

Dropout learning [28] is the way to deal with this type of trouble. This is extremely simple yet very efficient. Some of the hidden units are randomly detached with a predetermined probability in each training iteration and the learning operation continues on a regular basis. Those arbitrary perturbations restrain the network excellently from learning spurious interdependence and generating complicated co-adaptation amongst hidden units.

The architecture averaging initiated by dropout attempts to assure that every hidden unit learns representations of features that are primarily encouraging to achieve the appropriate classification result.

3. Motivation

The primary goal is to classify the audio by applying deep neural networks, which is a very successful approach in image classification. It has been a tremendous success in image classification and other related tasks including summary generation, object recognition and object detection, and so on, but a lot of things are not explored for audio classification and other problems like audio based search, keyword spotting.

The insight of whether an audio fragment has a particular class (e.g., Siren, drilling) gives important information involved in the fragment without natural language understanding. Not only that, the information of whether an audio fragment consists of speech or some other background noise may help to save huge human labor to segment sound files.

4. Task description

Given a sound clip, the purpose is to categorize/classify the sound fragments into a set of predefined category/class. Generally, it is a multilabel classification as each fragment may consist of overlapping segments of sound from multiple classes.

Mathematically,

Given $M = m_1, m_2, m_3. \ldots \ldots ..m_t$,

The task is to create labels $N = n_1, n_2, n_3. \ldots \ldots \ldots .n_t$ such that each n_i belongs to C

where C is the set of all classes.

We extracted mel and chroma features such as Mfcc, Mel spectrogram, chroma–stft, chroma–cqt and chroma-cens of the sound clip. Thus, the problem here is to classify these mel and chroma features of the given sound fragment into a set of defined classes.

5. Dataset

The UrbanSound8k dataset is used to examine the proposed neural model. It has 8732 categorized aural fragments of urban sounds of 10 categories like Air-conditioner, car-horn, children-playing, dog-bark, drilling, engine-idling, gun-shot, jackhammer, siren, and street-music. Most audio fragments have a length of less than or equal to 4 seconds. The categories are drawn from urban aural/acoustic taxonomy hooked up at the high frequency with which they are present within the noise complaints just as driven from the collection contributed by New York City's 311 service. There might be a possibility to present other sources in the fragment along with the labelled source because these are absolute natural field recordings. The sound source files have been annotated manually. The sound source files of urban sounds are in wav format. This data collection is considerably challenging one as most of the categories are extremely confusing, even to humans, such as jack hammer and drilling or air conditioner and engine idling because of high similarity in timbre and also the categories like children playing and street music has complex harmonic tones. The data collection is segmented into 10 folds for cross validation. The dataset is downloadable from [29].

6. Experimental results and discussion

Let us consider that the input data is f and kernel is h. The indices of rows and columns of the output matrix are denoted as m * n. So, the feature map values are calculated using the given formula

$$G[m,n] = f * h[m,n] = \sum_j \sum_k h[j,k] f[m-j, n-k]$$

For example, if we process a convolution across a 6x6 object with a 3x3 kernel, we will have a 4x4 feature map. It is because there are only 16 distinct positions in which we would locate our filter within the object. The feature map generated from the sound source is given as input to CNN architecture which follows the pipeline shown below. The model is trained on the discussed dataset.

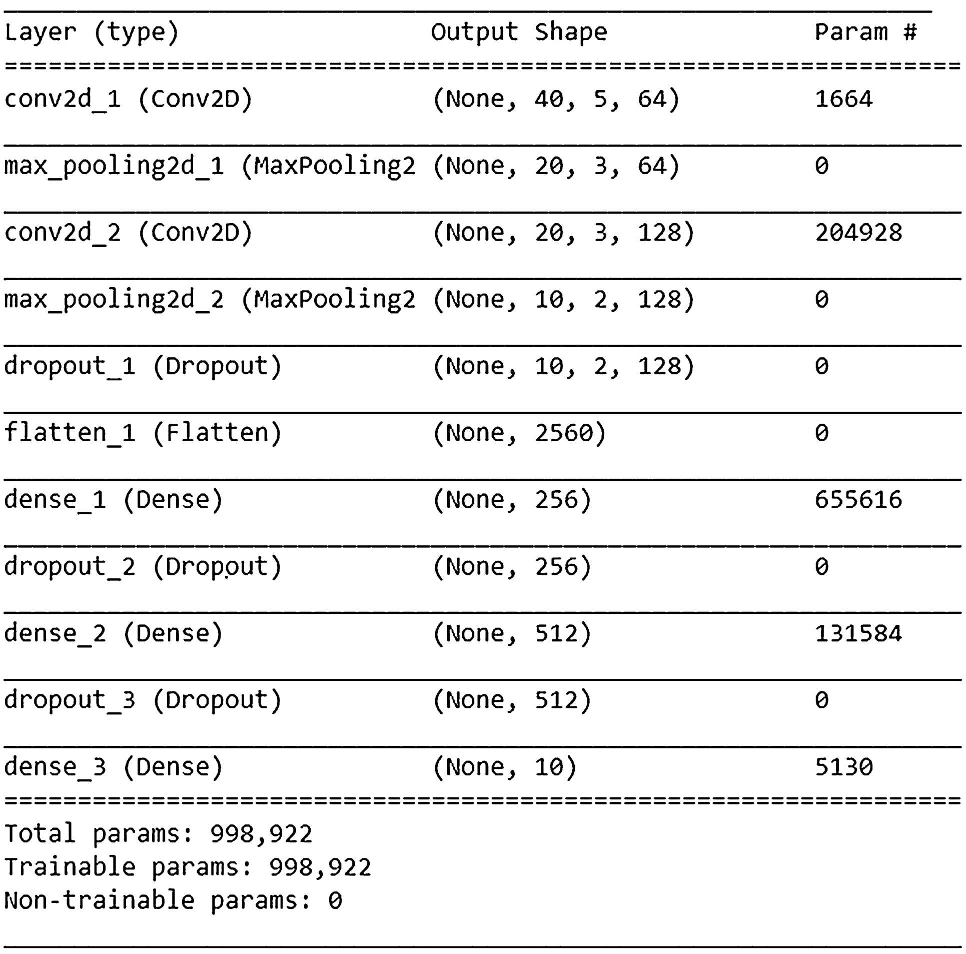
Dropout regularization, Adam optimizer [30] that robustly tunes the step size for each dimension, and ReLU for reduced computational cost [31] are used. Hyper parameters like batch size, epoch, and dropout is tuned whenever possible. The model is trained on the server with Processor Intel (R) Xeon (R) CPU E5-2609 V4 @1.70GHz and RAM 32.0 GB. The figure 2 shows the full structure for the JK-CNN network.

The table 1 briefs the mean accuracy achieved by the JK-CNN, along with other approaches examined on the urbanSound8k dataset. The JK-CNN achieved a validation accuracy of 76%, that is the topmost neural-based accuracy for the data set being examined.

The convolution result is obtained by the given formula

$$O_c = \frac{I - k + 2P}{s} + 1$$

Here, O_c is the output convolution, I is the input height, k is filter size, p is padding and s is stride.



```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_1 (Conv2D)            (None, 40, 5, 64)         1664
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 20, 3, 64)         0
_________________________________________________________________
conv2d_2 (Conv2D)            (None, 20, 3, 128)        204928
_________________________________________________________________
max_pooling2d_2 (MaxPooling2 (None, 10, 2, 128)        0
_________________________________________________________________
dropout_1 (Dropout)          (None, 10, 2, 128)        0
_________________________________________________________________
flatten_1 (Flatten)          (None, 2560)              0
_________________________________________________________________
dense_1 (Dense)              (None, 256)               655616
_________________________________________________________________
dropout_2 (Dropout)          (None, 256)               0
_________________________________________________________________
dense_2 (Dense)              (None, 512)               131584
_________________________________________________________________
dropout_3 (Dropout)          (None, 512)               0
_________________________________________________________________
dense_3 (Dense)              (None, 10)                5130
=================================================================
Total params: 998,922
Trainable params: 998,922
Non-trainable params: 0
_________________________________________________________________
```

Figure 2. JK-CNN Network tructure.

Table 1. Accuracies on the urban*Sound8k.*

	Accuracy	
Classifiers	Neural network	Non-neural network
JK-CNN+Mfcc+Melspectrogram+chroma-stft+ chroma-cqt+chroma-cens(Our work)	**76**	
MCLNN+Melspectrogram [33]	73.3	
Piczak-CNN+Melspectrogram [23]	73.1	
S&B-CNN+Melspectrogram without augmentation [19]	73.0	
CNN+DenseNet161+Dilation(without multiscale) [36]	72.53	
VGG+log-mel spectrogram [37]	59.51	
CNN+ResNets(with dilation) [37]	70.8	
CNN+DenseNets(with dilated multiscale) [36]	71.8	
CNN+ResNets(without dilation) [37]	69.7	
RBF-SVM+MFCC [29]	68.0	
RandomForest +Spherical KMeans+PCA+Melspectrogram [9]		73.7

In this model, padding is same padding applied to retain the output size as equal size as the input size.

$$\text{Padding} = P = \frac{k-1}{2}$$

Stride 1 is used which involves two operations. They are element-wise multiplication and summation. Max pooling is chosen as pooling function since an environment aural mostly heterogenous and this work is to classify sound exactly. Maxpool would be helpful in this kind of identification rather average pooling.

$$O_p = \frac{w-k}{s} + 1$$

Here, $\mathrm{O_p}$ is output pooling, w is weight, k is filter size and s is stride.

In Salamon [9], the highest non-neural based accuracy that is 73.7% stated. He examined random forest as a classifier practiced over a dictionary developed by employing spherical Kmeans [32]. To compare the JK-CNN with other CNN models, the proposed CNN used less than one million parameters trained over five features of 193 shape over spectrogram transformation. The other important two CNN models proposed by piczak [23] and MCLNN [33] adopted the spectrogram transformation for their models where piczak-CNN [23] model was trained using a separate spectrogram and delta channel. He examined with two fragment sizes derived from the spectrogram. The short fragment has 41 frames and 101 frames in the long fragment. The topmost accuracy attained [23] was 73.1% in a long fragment for training the PiczakCNN Model. The CNN model proposed by [23] comprised of two 80 filters convolution layers, 2 pooling layers, then 2 entirely connected 5000 neuron layers each and at last SoftMax output. The number of trained parameters exceed 25 million. The other model proposed by Fady [33] used the spectrogram transformation as same as piczak [23]. Column wise concatenated spectrogram and deltas producing a frame size of 120 features. Fady [33] used Parametric Rectifier Linear Unit (PReLU [34]) as an activation function. Followed by an overall mean pooling layer [35], the two masked layers used 3 million control parameters trained over 65 frames fragments. He achieved a 73.3% accuracy.

The CNN model proposed by salamon [9] applied lesser parameters than fady [33]. [9] used an augmentation stage by applying various modifications to the input signal such as time stretching, dynamic range compression and more to raising up the dataset and as a consequence improve the generalization of the model. Piczak [23] stated that there is no performance improvement, though they exploit augmentation on the urbansound8k dataset. So, in this we did not use augmentation because it is not an appropriate criterion to measure the performance of JK-CNN over other different models.

6.1 *Visualize training model history*

These plots generally give some insights about the training of the model like

- Model's training speed of convergence over epochs.
- Whether the model may have converged earlier.

Whether the model may be over-learning the training data.

Figure 3 shows clearly that the model could perhaps be trained a little more as the trend for curve, i.e., accuracy on train and test sets are still rising for the last few epochs.

One more thing is that it is transparently visible that the model has not over learned the training dataset, which reveals the comparable skill on both training and test datasets.

The given figure 4 shown the comparable performance on both train and validation datasets. A confusion matrix (or confusion table) shows a more detailed breakdown for each class of correct and incorrect classifications. The matrix rows correspond to labels of ground truth, and the columns are the prediction. There is always confusion between Airconditioner, Drilling, Engine idling and Jack hammer. It is due to the existence of same low tones among these classes. The Confusion Matrix in figure 5 reveals the classification detail of JK-CNN model architecture.

The confusion matrix for the JK-CNN model achieves noticeably better than [19] in spotting specific categories like Air Conditioner, children playing, dog park, jack hammer and street music, but at the mean time behaves somewhat poor for aurals like gunshot and car horn. Compared to [30], JK-CNN model very much decreases the confusion between the most challenging aural classes such as air conditioner and drilling, air conditioner and engine idling as well as engine idling and jack hammer. Finally, in some categories, the confusion still persists

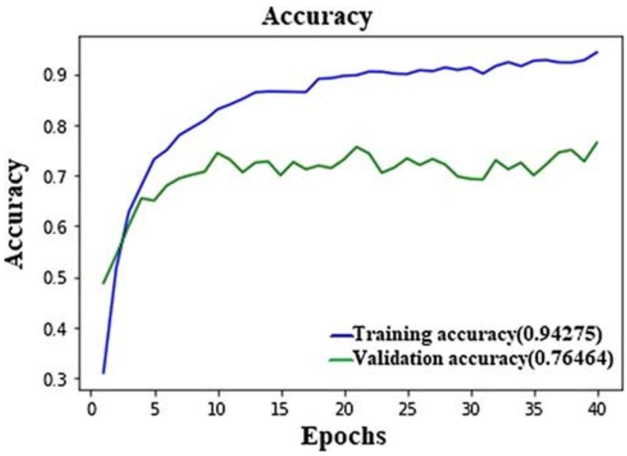


Figure 3. Plot of Accuracy on Training and Test Dataset.

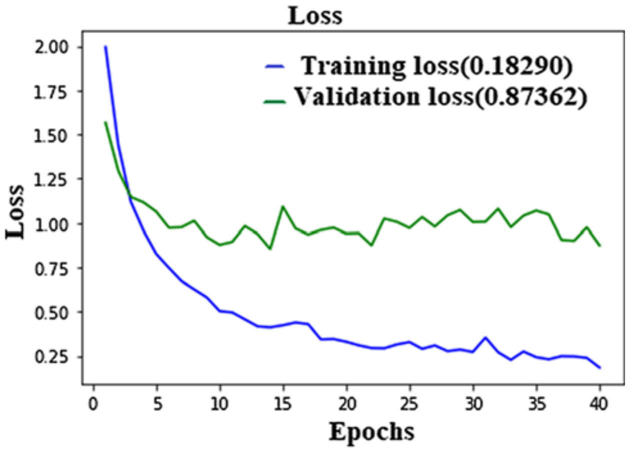


Figure 4. Plot of Loss on Training and Test Dataset.

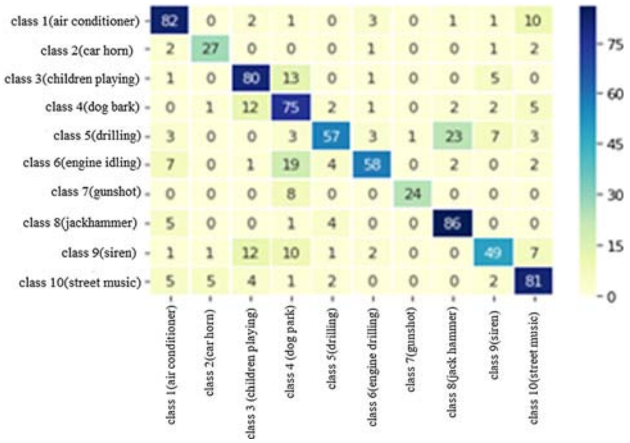


Figure 5. Confusion Matrix for JK-CNN.

between engine idling and dog bark due to short scale temporal nature.

The average precision score compiles a precision recall curve just as the weighted average of accuracy achieved at particular threshold, with the increment in recall from the prior threshold being used as the weight. This score estimates the average precision from prediction score. This value lies between 0 and 1. In a single term, Average precision is the fraction of positive samples. For this JK-CNN model, the average precision score is 0.82 and the corresponding curve is shown in figure 6.

Precision recall curve is to show the result of the classifier. For multi-class classifier, this curve is extended to draw for each and every class, but a precision recall curve could be further drawn by considering the specific factor of the class indicator matrix as a micro-averaging. For this JK-CNN model, the multi class curve is shown in figure 7.

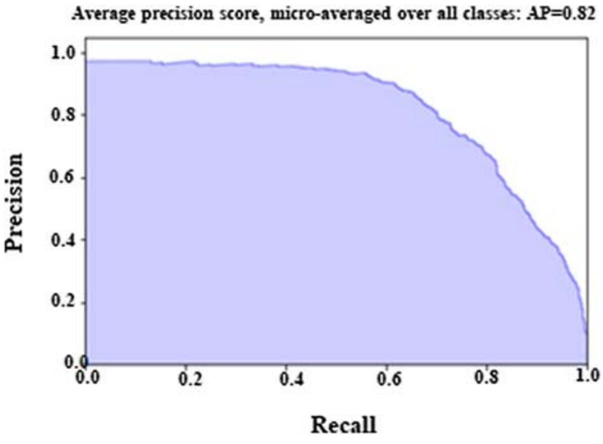


Figure 6. Average Precision Recall Score for JK-CNN.

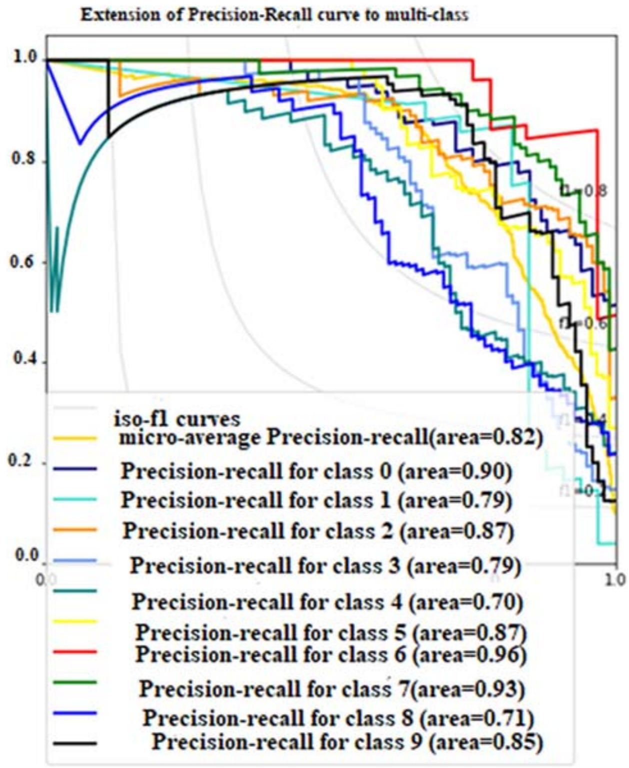


Figure 7. Precision-Recall curve for each class and iso-f1 curves of proposed JK-CNN.

7. Concluding remarks

Deep Convolutional Network that perform well on low level representation of sound clips is proposed. The proposed architecture has two convolutional layers along with MaxPooling and three fully interconnected layers. The results reveal that our model outperforms various other CNN models and achieved the highest accuracy of 76%, competitive with other CNNs. The confusion across the aural of UrbanSound environmental dataset is analyzed. The model's Average Precision Recall score and Precision Recall value for each class is shown. In future, the trained model is to be applied on other kind of Multimedia data especially video data.

References

[1] Chu S, Narayanan S, Jay Kuo C-C and Mataric M J 2006 Where am i scene recognition for mobile robots using audio features. In: *2006 IEEE International Conference on multimedia and expo*, pp. 885–888

[2] Abu-El-Quran A R, Goubran R A and Chan A D C 2006 Security monitoring using microphone arrays and audio classification. *IEEE Trans. Instrum. Meas.* 55(4):1025–1032

[3] Louradour J, Rouas L and Ambellouis S 2006 Audio events detection in public transport vehicle. In: *2006 IEEE Intelligent Transportation Systems Conference*, pp. 733–738

[4] Ntalampiras S, Potamitis I and Fakotakis N 2009 An adaptive framework for acoustic monitoring of potential hazards. *EURASIP J. Audio Speech Music Process.* 2009:13

[5] Ghiurcau M V, Rusu C, Bilcu R C and Astola J 2012 Audio based solutions for detecting intruders in wild areas. *Signal Process.* 92: 829–840

[6] Istrate D, Castelli E, Vacher M, Besacier L and Serignat J-F 2006 Information extraction from sound for medical telemonitoring. *IEEE Trans. Inf. Technol. Biomed.* 10(2):264–274

[7] Maijala P, Shuyang Z, Heittola T and Virtanen T 2018 Environmental noise monitoring using source classification in sensors. *Appl. Acoust.* 129:258–267

[8] Salamon J, Jacoby C and Bello J P 2014 A dataset and taxonomy for urban sound research. In: *Proceedings of the 22Nd ACM International Conference on Multimedia*, MM '14, pp. 1041–1044, New York, NY, USA ACM

[9] Salamon J and Bello J P 2015 Unsupervised feature learning for urban sound classification. In: *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 171–175

[10] Ye J, Kobayashi T and Murakawa M 2016 Urban sound event classification based on local and global features aggregation. *Appl. Acoust.* 117: 11

[11] Chu S, Narayanan S and Jay Kuo C-C 2009 Environmental sound recognition with time–frequency audio features. *IEEE Trans. Audio Speech Lang. Process.* 17(6):1142–1158

[12] Ellis D P W and Lee K 2004 Minimal-impact audio- based personal archives. In: *Proceedings of the 1st ACM Workshop: on Continuous Archival and Retrieval of Personal Experiences*, pp. 39–47

[13] Heittola T, Mesaros A, Eronen A and Virtanen T 2010 Audio context recognition using audio event histograms. In: *2010 18th European IEEE Signal Processing Conference*, pp. 1272–1276

[14] Chaudhuri S and Raj B 2013 Unsupervised hierarchical structure induction for deeper semantic analysis of audio. In: *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 833–837

[15] Radhakrishnan R, Divakaran A and Smaragdis A 2005 Audio analysis for surveillance applications. In: *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, 2005*, pp. 158–161

[16] Cai R, Lu L, Hanjalic A, Zhang H-J and Cai L-H 2006 A flexible framework for key audio effects detection and auditory context inference. *IEEE Trans. Audio Speech Lang. Process.* 14(3):1026–1039

[17] Heittola T, Mesaros A, Eronen A and Virtanen T 2013 Context-dependent sound event detection. *EURASIP J. Audio Speech Music Process.* 2013(1):1

[18] Cotton C V and Ellis D P W 2011 Spectral vs. spectro-temporal features for acoustic event detection. In: *2011 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, pp. 69–72

[19] Salamon J and Bello J P 2016 Deep convolutional neural networks and data augmentation for environmental sound classification. *CoRR*. abs/1608.04363

[20] Piczak K J 2015 Esc: dataset for environmental sound classification. In: *Proceedings of the 23rd ACM International Conference on Multimedia*, pp. 1015–1018. ACM

[21] Mesaros A, Heittola T and Virtanen T 2016 database for acoustic scene classification and sound event detection. In: *2016 24th European IEEE Signal Processing Conference (EUSIPCO)*, pp. 1128–1132

[22] Krizhevsky A and Hinton H 2012 GE: ImageNet classification with deep CNN. In: *Advances in Neural Information Processing Systems*, pp. 1097–1105

[23] Piczak K J 2015 Environmental sound classification with convolutional neural networks. In: *2015 IEEE 25th International Workshop on Machine Learning for Signal Processing (MLSP)*, pp. 1–6

[24] Temko A, Malkin R, Zieger C, Macho D, Nadeu C and Omologo M 2006 Clear evaluation of acoustic event detection and classification systems. In: *International Evaluation Workshop: on Classification of Events, Activities and Relationships*, pp. 311–322. Springer

[25] Choi I, Kwon K, Bae S H and Kim N S 2016 DNN-based sound event detection with exemplar-based approach for noise reduction. In: *Proceedings of the Detection and Classification of Acoustic Scenes and Events Workshop*, pp. 16–19

[26] Glorot X, Bordes A and Bengio Y 2011 Deep sparse rectifier neural networks. In: *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics, Volume 15 of Proceedings of Machine Learning Research*, pp. 315–323. PMLR, 11–13

[27] Xu B, Wang N, Chen T and Li M 2015 Empirical evaluation of rectified activations in convolutional network. *CoRR*. abs/1505.00853

[28] Hinton G E, Srivastava N, Krizhevsky A, Sutskever I and Salakhutdinov R R 2012 Improving neural networks by preventing co-adaptation of feature detectors. *CoRR*. abs/1207.0580. cite

[29] Salamon J, Jacoby C and Bello J P 2014 A dataset and taxonomy for urban sound research. In: *22nd ACM International Conference on Multimedia (ACM-MM'14)*, pp. 1041–1044, Orlando, FL, USA

[30] Kingma D and Ba J 2014 Adam: a method for stochastic optimization. In: *International Conference on Learning Representations*, 12 2014

[31] Zeiler M D, Ranzato M, Monga R, Mao M, Yang K, Le Q V, Nguyen P, Senior A, Vanhoucke V, Dean J and Hinton GE 2013 On rectified linear units for speech processing. In: *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 3517–3521

[32] Dhillon I S and Modha D S 2001 Concept decompositions for large sparse text data using clustering. *Mach. Learn.* 42(1):143–175

[33] Medhat F, Chesmore D and Robinson J 2017 Masked conditional neural networks for environmental sound classification. In: *SGAI Conference*

[34] He K, Zhang X, Ren S and Sun J 2015 Delving deep into rectifiers: surpassing human-level performance on imagenet classification. *CoRR.* abs/1502.01852

[35] Lin M, Chen Q and Yan S 2013 Network in network. *CoRR.* abs/1312.4400

[36] McMahan B and Rao D 2018 Listening to the world improves speech command recognition. In: *Thirty-Second AAAI Conference on Artificial Intelligence (2018)*

[37] Kumar Y, Vyas M, Garg. Department of computer science, Automatic Speech Recognition, *From image classification to audio classification*, saurabhgarg1996.github.io