A CNN Approach for Audio Classification in Construction Sites

Alessandro Maccagno[1], Andrea Mastropietro[1], Umberto Mazziotta[1], Michele Scarpiniti[2], Yong-Cheol Lee[3], and Aurelio Uncini[2]

[1] Department of Computer, Control and Management Engineering,
Sapienza University of Rome, Italy,
{maccagno.1653200,mastropietro.1652886,mazziotta.1647818}@studenti.uniroma1.it,
[2] Department of Information Engineering, Electronics and Telecommunications,
Sapienza University of Rome, Italy
{michele.scarpiniti,aurelio.uncini}@uniroma1.it,
[3] Department of Construction Management,
Louisiana State University, Baton Rouge, USA
yclee@lsu.edu

Abstract. Convolutional Neural Networks (CNNs) have been widely used in the field of audio recognition and classification, since they often provide positive results. Motivated by the success of this kind of approach and the lack of practical methodologies for the monitoring of construction sites by using audio data, we developed an application for the classification of different types and brands of construction vehicles and tools, which operates on the emitted audio through a stack of convolutional layers. The proposed architecture works on the mel-spectrogram representation of the input audio frames and it demonstrates its effectiveness in environmental sound classification (ESC) achieving a high accuracy. In summary, our contribution shows that techniques employed for general ESC can be also successfully adapted to a more specific environmental sound classification task, such as event recognition in construction sites.

Keywords: Deep learning, Convolutional neural networks, Audio processing, Environmental sound classification, Construction sites.

1 Introduction

In last years, many research efforts have been made towards the event classification of audio data, due to the availability of cheap sensors [1]. In fact, systems based on acoustic sensors are of particular interest for their flexibility and cheapness [2]. When we consider generic outdoor scenarios, an automatic monitoring system based on a microphone array would be an invaluable tool in assessing and controlling any type of situation occurring in the environment [3]. This includes, but is not limited to, handling large civil and/or military events. The idea in these works is to use Computational Auditory Scene Analysis (CASA) [4], which involves Computational Intelligence and Machine Learning techniques, to recognize the presence of specific objects into sound tracks. This last problem is a notable example of Automatic Audio Classification (AAC) [5], the task of automatically labeling a given audio signal in a set of predefined classes.

Getting into the more specific field of environmental sound classification (ESC) in construction site, the closest attempts have been performed by Cheng et al. [6], who used Support Vector Machines (SVM) to analyze the activity of construction tools and equipment. Recent applications of AAC have also been addressed to audio-based construction sites monitoring [7–9], in order to improve the construction process management of field activities. This approach is revealing itself as a promising method and a supportive resource for unmanned field monitoring and safety surveillance that leverages construction project management and decision making [8, 9]. More recently, several studies extend these efforts to more complicated architectures exploiting Deep Learning techniques [10].

In the literature, it is possible to find several instances of successful applications in the field of environmental sound classification that make use of deep learning. For example, in the work of Piczak [11], the author exploits a 2-layered CNN working on the spectrogram of the data to perform ESC, reaching an average accuracy of 70% over different datasets. Other approaches, instead of using handcrafted features such as the spectrogram, perform end-to-end environmental sound classification obtaining higher results with respect to the previous ones [12, 13].

Inspired and motivated by the MelNet architecture described by Li et al. [14], which has been proven to be remarkably effective in environmental sound classification, the aim of this paper is to develop an application able to recognize vehicles and tools used in construction sites, and classify them in terms of type and brand. This task will be tackled with a neural network approach, involving the use of a Deep Convolutional Neural Network (DCNN), which will be fed with the mel spectrogram of the audio source as input. The classification will be carried on five classes extracted from audio files collected in several construction sites, containing in situ recordings of multiple vehicles and tools. We demonstrate that the proposed approach for ESC can obtain good results (average accuracy of 97%) to a very specific domain as the one of construction sites.

The rest of this paper is organized as follows. Section 2 describes the proposed approach used to perform the sound classification. Section 3 introduces the experimental setup, while Section 4 shows the obtained numerical results. Finally, section 5 concludes the paper and outlines some future directions.

2 The proposed approach

CNNs are a particular type of neural networks, which use the *convolution* operation in one or more layers for the learning process. These networks are inspired by the *primal visual system*, and are therefore extensively used with image and video inputs [10]. A CNN is composed by three main layers:

- **Convolutional Layer**: The convolutional layer is the one tasked with applying the convolution operation on the input. This is done by passing a filter (or *kernel*) over the matricial input, computing the convolution value, and using the obtained result as the value of one cell of the output matrix (called *feature map*); the filter is then shifted by a predefined *stride* along its dimensions. The filters parameters are trained during the training process.

– **Detector layer**: In the detector layer, the output of the convolution is passed through a nonlinear function, usually a ReLU function.

– **Pooling layer**: The pooling layer is meant to reduce the dimensionality of data by combining the output of neuron clusters at one layer into one single neuron in the subsequent layer.

The last layer of the network is a fully connected one (a layer whose units are connected to every single unit from the previous one), which outputs the probability of the input to belong to each of the classes.

CNNs in a machine learning system show some advantages with respect to traditional fully connected neural networks, because they allow sparse interactions, parameters sharing and equivariant representations.

The reasons why we used CNNs in our approach is due to the intrinsic nature of audio signals. CNNs are extensively used with images and, since the spectrum of the audio is an actual picture of the signal, it is straightforward to see why CNNs are a good idea for such kind of input, being able to exploit the adjacency properties of audio signals and recognize patterns in the spectrum images that can properly represent each one of the classes taken into consideration.

The proposed architecture consists in a DCNN composed of eight layers, as shown in Fig. 1, that is fed with the mel spectrogram extracted from audio signals and its time derivative. Specifically, we have as input a tensor of dimension $60 \times 2 \times 2$ that is a couple of images representing the spectrogram and its time derivative: 60 is the number of mel bands, while 2 is the number of time buckets. Then, we have five convolutional layers, followed by a dense fully connected layer with 200 units and a final softmax layer that performs the classification over the 5 classes. The structure of the proposed network is summarized in the following Table 1, and it can be graphically appreciated in Fig. 1.

Layer	**Input Shape**	**Filters**	**Kernel Size**	**Strides**	**Output Shape**
Conv1	[batch, 60, 2, 2]	24	(6,2)	(1,1)	[batch, 60, 2, 24]
Conv2	[batch, 60, 2, 24]	24	(6,2)	(1,1)	[batch, 60, 2, 24]
Conv3	[batch, 60, 2, 24]	48	(5,1)	(2,2)	[batch, 30, 1, 48]
Conv4	[batch, 30, 1, 48]	48	(5,1)	(2,2)	[batch, 15, 1, 48]
Conv5	[batch, 15, 1, 48]	64	(4,1)	(2,2)	[batch, 8, 1, 64]
Flatten	[batch, 8, 1, 64]	–	–	–	[batch, 512]
Dense	[batch, 512]	–	–	–	[batch, 200]
Output - Dense	[batch, 200]	–	–	–	[batch, 5]

Table 1. Parameters of the proposed DCNN architecture.

All the layers employ a ReLu activation function except for the output layers which uses a Sofmax function. The optimizer chosen for the network is an Adam Optimizer [15], with the a learning rate set to 0.0005. Such value was chosen by performing a grid search in the range [0.00001, 0.001]. Moreover, a dropout strategy, with a rate equal to 30%, has been used in the dense layer.

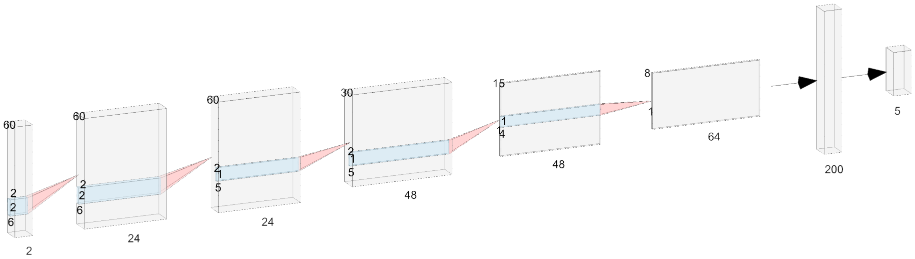

Fig. 1. Graphical representation of the proposed architecture.

Regarding the setting of other hyper-parameters, different strategies were adopted. For the batch size, a grid search was used to determine the most appropriate values. The filter size and the stride were set reasonably according to the input size. Small filters were adopted such to capture small, local and adjacent features that are typical of audio data. Lastly, to prevent the network depth from either exploding in size, adding unnecessary complexity for no actual return, or not being high enough, therefore returning substandard results, we decided to use the same amount of layers as other related works, such as the one in [14], as a baseline. Variations on this depth have not shown appreciable improvements on the overall effectiveness of the networks classification, so it has been kept unchanged.

2.1 Spectrogram Extraction

The proposed DCNN uses, as its inputs, the mel spectrogram that is a version of the spectrogram where the frequency scale has been distorted in a perceptual way, and its time derivative.

The technique used to extract the spectrogram from the sample is the same used by Piczak [11], via the Python library `librosa`[4] . The frames were resampled to 22,050 Hz, then a window of size 1024 with hop-size of 512 and 60 mel bands has been used. A mel band represents an interval of frequencies which are perceived to have the same pitch by human listeners. They have been found to be performing in speech recognition.

With this parameters, and the chosen length of 30 ms for the frames (see next sections), we obtain a small spectrogram of 60 rows (bands) and 2 columns

[4] Available at: https://librosa.github.io/

(buckets). Then, using again `librosa`, we compute the derivative of the spectrogram and we overlap the two matrices, obtaining a dual channel input which is fed into the network.

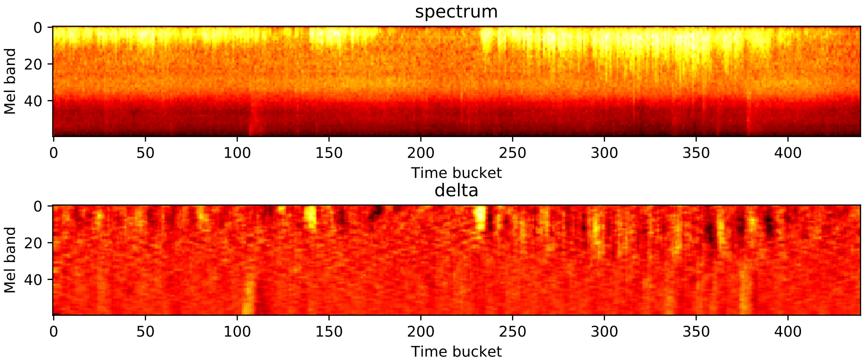

Fig. 2. Example of a log-mel spectrogram extracted from a fragment along with its derivative. On the abscissa we find the time buckets, each of which representing a sample about 23 ms long, while on the ordinates the log-mel bands. Since our fragments are 30 ms long, the spectrogram we extract will contain 2 buckets.

3 Experimental setup

3.1 Dataset

The authors collected audio data of equipment operations in several construction sites consisting diverse construction machines and equipments. Unlike artificially built datasets, when working with real data different problems arise, such as noise due to weather conditions and/or workers talking among themselves. Thus, we focused our work on the classification of a reduced number of classes, specifically *Backhoe JD50D Compact*, *Compactor Ingersoll Rand*, *Concrete Mixer*, *Excavator Cat 320E*, *Excavator Hitachi 50U*. Classes which did not have enough usable audio (too short, excessive noise, low quality of the audio) were ignored for this work. The activities of these machines were observed during certain periods, and the audio signals generated were recorded accordingly. A Zoom H1 digital handy recorder has been used for data collection purposes. All files have been recorded by using a sample rate of 44,100 Hz and a total of about one hour of sound data (eight different files for each machine) has been used to train the architecture.

3.2 Data Preprocessing

In order to feed the network with enough and proper data, each audio file for each class is segmented into fixed length frames (the choice of the best frame size is described in the experiment section). As first step, we split the original audio files into two parts, training samples (70% of the original length) and test samples (30% of the original length). This is done to avoid testing the network on data used previously to train the network, as this would cause the network to overfit and give misleading results.

Then, we perform data augmentation by splitting the files into smaller segments of 30 ms, each of which overlaps the subsequent one by 15 ms. We then compute the Root Mean Square (RMS) of every signal of these frames, and drop the ones with too small power with respect to the average RMS of the different segments, in order to remove the frames which contain mostly silence.

After that, the dataset is balanced by taking N samples for each class, where N is the number of elements contained in the class with the least amount of samples. In this way, we avoid the problem of having certain classes with an abnormal number of usable audio segments being potentially either over-represented or under-represented and negatively impacting the training of the model, especially due to the presence of multiple models of the same vehicle.

Using the Python library `librosa`, we extract the waveform of the audio tracks from the audio samples and, using the same library, we generate the log-scaled mel spectrogram [16] of the signal and its time derivative that will be the input to the network.

Numerical results have been evaluated in terms of accuracy, recall, precision and F1 score [17].

4 Numerical Results

4.1 Selecting the frame size

A sizeable amount of time was spent into finding the proper length for the audio segments. This is of crucial importance since, if the length is not adequate, the network will not be able to learn proper features that clearly characterize the input. Hence, in order to select the most suitable length, we generated different dataset variants by splitting the audio using different frame lengths, and we subsequently trained and tested different models on the differently-sized datasets. The testing results in terms of overall accuracy are show in Fig 3.

As we can see, with a smaller frame size better results are obtained, while we notice a drop as the size increases. It is also interesting to observe that with a very large frame size the accuracy tends to slightly improve again. However, the use of long frames does not lead to anything interesting since the network may tend to learn an ensemble of the signal that is not significant and useful to work in fast-response applications (hazard detection, activity monitoring, etc.). Finally, the optimal frame size is obtained by selecting a duration of 30 ms, since it led not only to achieve a high accuracy but also a larger number of samples.

In order to properly test the network we performed a K-fold cross validation, with $K = 5$. The results of the classification are shown in the next subsection.

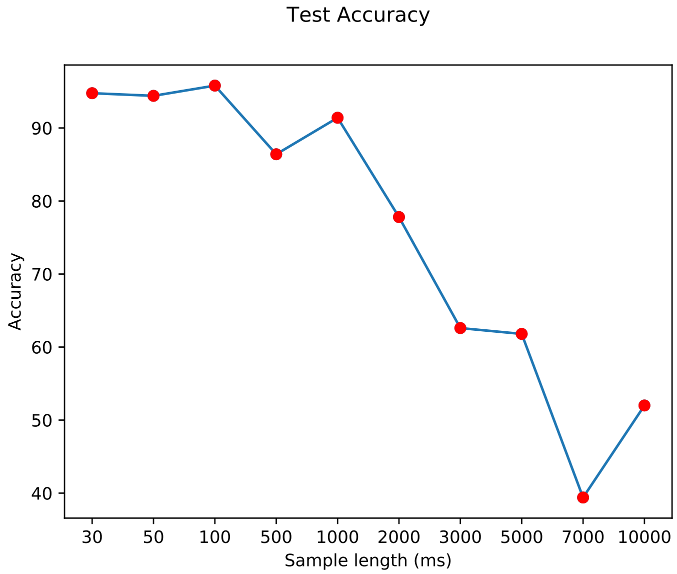

Fig. 3. Overall classification accuracy according to different sample sizes of the audio frames.

4.2 Classification Results

As just stated, a 5-fold cross validation was performed and the results are shown in Table 2. The dataset was split into training set and validation set (80% – 20%) for each fold.

Class	**Accuracy**	**Recall**	**Precision**	**F1**
Backhoe JD50D Compact	98.52	97.23	95.54	96.34
Compactor Ingersoll Rand	98.73	97.89	95.71	96.76
Concrete Mixer	99.21	98.49	97.58	98.03
Excavator Cat 320E	99.19	97.34	98.60	97.96
Excavator Hitachi 50U	98.99	97.82	97.16	97.49
All classes	**97.08**	**97.34**	**97.30**	**97.32**

Table 2. 5-Fold cross validation classification results (in %).

As we can notice, the network achieves very high results in all the metrics, demonstrating its effectiveness in this particular domain. Even though our classes also include vehicles of the same type (we have two excavators and a backhoe, which is a kind of excavator as well), such classes are discriminated in a very clear and accurate way as the net recognizes also the brand of the machine.

After having performed the cross validation, we trained the network again on the original version of the dataset (training set 70% and test set 30%) and tested it. The way the network learns can be seen in Fig. 4; the learning is actually really fast as we see that high overall accuracy values are reached within few epochs and thus the convergence is rapid. The accuracy results obtained are shown in the confusion matrix in Fig. 5. From this figure, it is clear that all classes are well correctly recognized, since the accuracy is always higher than 95%. The class with worst result is the Excavator Cat 320E that performs at 95% of accuracy.

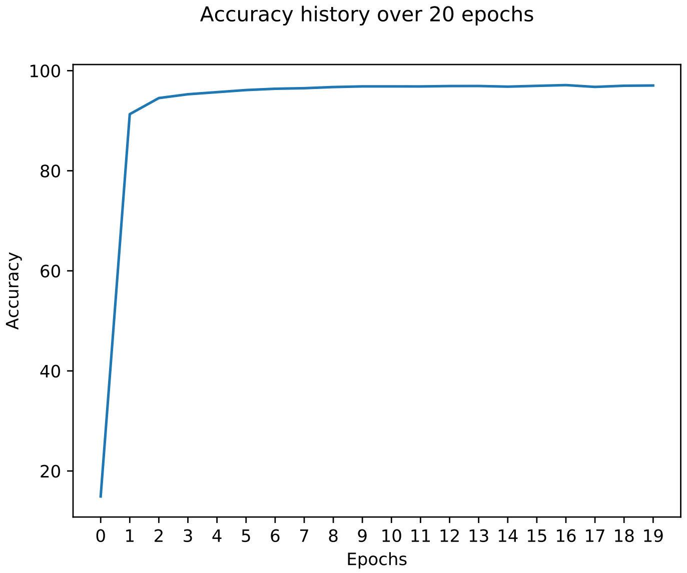

Fig. 4. Overall accuracy obtained on the test set.

As a comparison, we perform classification with other five state-of-the-art classifiers, namely Random Forest, Multilayer Perceptron (MLP), k-NN and Support Vector Machine (SVM) [17]. These classifiers take into their inputs a set of 62 features extracted from audio signals. All details, features and parameters of the implemented classifiers can be found in [8]. Results of these considered approaches, averaged over the five classes, are shown in Table 3. From this table, we can see that the state-of-the-art approaches always produce worse results than those of the proposed architecture, shown in the last line of Table 3. This is due to the powerful feature representation and discrimination of the used DCNN and the mel spectrogram signal representation.

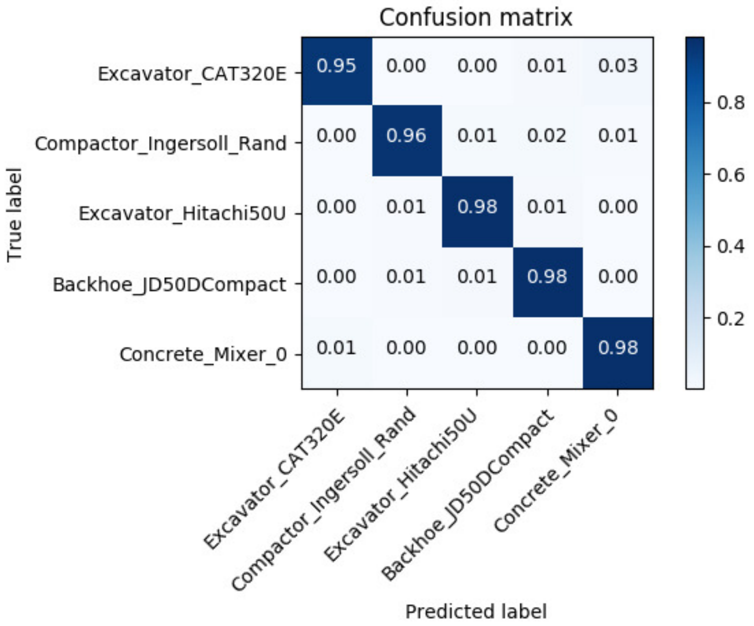

Fig. 5. Confusion matrix obtained by the proposed approach.

4.3 Prediction

The proposed approach can be used to promptly predict the active working vehicles and tools. In fact, with such an approach, project managers will be able to remotely and continuously monitor the status of workers and machines, investigate the effective distribution of hours, and detect issues of safety in a timely manner.

In order to predict a new sample in input, the recorded audio file is split into frames as described above. Every frame will be classified as belonging to one of the classes and the audio track will be labeled according to the majority of the

Approach	**Accuracy**	**Recall**	**Precision**	**F1**
Random Forest	93.16	93.21	93.40	93.30
MLP	91.06	93.20	91.34	92.26
k-NN	85.28	85.32	86.04	85.68
SVM	83.66	83.75	84.63	84.19
DCNN (**Proposed**)	97.08	97.34	97.30	97.32

Table 3. Averaged results of compared classifiers (in %).

labels among all the frames. In this way, we can also see what is the probability for the input track to belong to each of the classes.

5 Conclusions and Future Work

In this paper, we demonstrated that it is possible to apply a neural approach already tested in environmental sound classification to a more specific and challenge domain, that is the one of construction sites, obtaining rather high results. Such architecture works with small audio frames and, for practical applications, the ability to perform a classification using very short samples can lead to the possibility to use such network in time-critical applications in construction sites that require fast responses, such as hazard detection and activity monitoring.

Up to now, the proposed architecture was tested on five classes obtaining an accuracy of 97%. The idea is to try to increase the number of classes to include more tools and vehicles employed in building sites, in order to lead in the future to a more reliable and useful system. Moreover, the most interesting way to extend the work would be to try to combine more architectures in order to establish which kind of neural networks can help the audio classification in construction sites.

References

1. S. Scardapane, M. Scarpiniti, M. Bucciarelli, F. Colone, M. V. Mansueto, and R. Parisi, “Microphone Array Based Classification for Security Monitoring in Unstructured Environments,” AEÜ – International Journal of Electronics and Communications, Vol. 69, N. 11, pp. 1715–1723, November 2015.
2. E. Weinstein, K. Steele, A. Agarwal, and J. Glass, “LOUD: a 1020-node modular micro-phone array and beamformer for intelligent computing spaces,” Tech. rep. MIT/LCS Technical Memo MIT-LCS-TM-642, 2004.
3. B. Kaushik, D. Nance, and K. K. Ahuja, “A review of the role of acoustic sensors in the modern battlefield,” in *Proc. of the 11-th AIAA/CEAS Aeroacoustics Conference*, pp. 1–13, 2005.
4. D. Wang and G. J. Brown, *Computational auditory scene analysis: principles, algorithms, and applications*, Wiley-IEEE Press, 2006.
5. Z. Fu, G. Lu, K. M. Ting, and D. Zhang “A survey of audio-based music classification and annotation,” *IEEE Transactions on Multimedia*, Vol. 13, N. 2, pp. 303–319, 2011.
6. C.-F. Cheng, A. Rashidi, M. A. Davenport, and D. V. Anderson, “Activity analysis of construction equipment using audio signals and support vector machines,” *Automation in Construction*, Vol. 81, pp. 240–253, 2017.
7. T. Zhang, Y.-C. Lee, M. Scarpiniti, and A. Uncini, “A Supervised Machine Learning-Based Sound Identification for Construction Activity Monitoring and Performance Evaluation,” in *Proc. of 2018 Construction Research Congress (CRC 2018)*, New Orleans, Louisiana, USA, pp. 358–366, April 2–4, 2018.
8. Y.-C. Lee, M. Scarpiniti, and A. Uncini, “Advanced Sound Identification Classifiers Using a Grid Search Algorithm for Accurate Audio-based Construction Progress Monitoring,” submitted to *Journal of Computing in Civil Engineering*, 2019.

9. B. Sherafat, A. Rashidi, Y.-C. Lee, and C. R. Ahn, “A Hybrid Kinematic-Acoustic System for Automated Activity Detection of Construction Equipment,” *Sensors*, Vol. 19, N. 19, Paper 4286, Oct. 2019.
10. I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning.* MIT Press, 2016.
11. K. J. Piczak, “Environmental sound classification with convolutional neural networks,” in *2015 IEEE 25th International Workshop on Machine Learning for Signal Processing (MLSP)*, pp. 1–6, Sep. 2015.
12. Y. Tokozume and T. Harada, “Learning environmental sounds with end-to-end convolutional neural network,” in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2721–2725, March 2017.
13. Y. Xie, Y.-C. Lee, and M. Scarpiniti, “Deep Learning-Based Highway Construction and Maintenance Activities Monitoring in Night Time,” submitted to *Construction Research Congress (CRC 2020)*, Tempe, AZ, USA, March 8–10, 2020.
14. S. Li, Y. Yao, J. Hu, G. Liu, X. Yao, and J. Hu, “An ensemble stacked convolutional neural network model for environmental event sound recognition,” *Applied Sciences*, Vol. 8, N. 7, 2018.
15. D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” in *ICLR*, 2015.
16. E. B. Stevens, Stanley Smith; Volkmann; John & Newman, “A scale for the measurement of the psychological magnitude pitch,” 1937.
17. E. Alpaydin, *Introduction to Machine Learning*, Mit Press, 3rd Ed., 2014.