Avian Song Identification Using CNN

Md Raihan Uddin, Abu Asaduzzaman, Rebecca Soza, and Chaz Minkler
Department of Electrical and Computer Engineering
Wichita State University
Wichita, Kansas, USA
mxuddin11@shockers.wichita.edu

Abstract—**Avian songs, a fundamental element of bird ecology, serves a crucial function in communication, mate selection, territorial dynamics, and many more. Precise identification of avian vocalizations is imperative for comprehending ecosystem health and facilitating efficient conservation endeavors. Conventional approaches encounter difficulties owing to the intricate nature of avian vocalizations, marked by a variety of song types, regional distinctions, and individual subtleties. Studies show that Machine Learning (ML) applications have the potential to address these issues accurately and conveniently. This study proposes a method to identify avians with their songs in real life environment using two different sizes of Convolutional Neural Network (CNN) models. The proposed methodology involves converting audio files into spectrograms, followed by training and validating on 40 different bird species. The conclusive prediction is derived from a dataset comprising 10-minute soundscapes, processed into 120 spectrograms with a 5-second interval each. According to the experimental findings, the integrated model demonstrates an accuracy of approximately 86% in forecasting the overall count of bird species within each soundscape.**

Keywords—Bird audio detection (BAD), CNN, Spectrogram, soundscape

I. Introduction

Birdsong is a cornerstone of avian ecology, intricately woven into communication, mate selection, and territorial dynamics [1]. The distinctive vocalizations of different bird species offer profound insights into the presence, distribution, and abundance of avian populations within ecosystems. Avian song analysis, a pivotal component of bioacoustics, not only contributes to scientific inquiry but also fosters public engagement through citizen science initiatives [2]. Variation in bird vocalizations serves as an early warning system for ecological disruptions caused by environmental changes [3].

Furthermore, avian song identification helps to predict the geographic range of various bird species. Conservationists can identify priority areas for protection and restoration efforts, enhancing the overall effectiveness of conservation strategies [4]. Understanding how bird species adapt to environmental changes is pivotal for predicting and mitigating the impacts of climate change on biodiversity. Thus, avian song identification emerges as a powerful tool that not only advances scientific knowledge but also provides a strategic foundation for proactive conservation measures. [5], [6].

As technological advancements continue to expand the horizons of animal behavior studies, the field of avian song identification has witnessed increased interest. However, this surge in attention is accompanied by a recognition of the complex challenges associated with accurate and efficient identification [7]. Avian vocalizations are inherently complex, characterized by a myriad of song types, regional variations, and individual nuances, presenting a formidable obstacle to precise identification. The traditional methods of avian song identification are challenging which involves a combination of manual analysis and automated techniques [8].

Recent advancements, particularly the application of Convolutional Neural Network (CNN) models, have shown promise in transforming the landscape of avian bioacoustics [9-14]. This paper builds upon existing research by offering a methodology in avian song identification. Two different sizes of CNN model are used to predict avian songs from 40 species. The dataset is sourced from xenocanto.org contains 10,955 samples. Validation accuracy for Model I and II is 72% and 70% respectively. At the end, 20 soundscape audio files are tested, and the model shows 86% accuracy in predicting 40 bird species.

The rest of the paper is organized as follows: Section II, the literature review, offers a comprehensive survey of current research and studies pertaining to the application of Machine Learning (ML) techniques in predicting avian songs and bird species and CNN classification method has been discussed in detail. The proposed methodology is explained in Section III, while Section IV & V details the conducted experiments and elucidates the results achieved respectively, highlighting the efficacy and accuracy of the developed models. Lastly, Section VI draws conclusions for the study by summarizing the research findings within the realm of ML-based avian song and bird species prediction.

II. Background Materials

A. Literature Review

Several research have been done to identify bird audio or instrument audio using ML algorithms or models. The summary of the methodology and performance of different ML models are listed in Table I.

Cakir et al. [10], explores Convolutional Recurrent Neural Network (CRNN) for detecting bird audio from a dataset of Xeno-canto, encompassing recordings across various acoustic environments. The CRNN method is evaluated on unseen data, achieving an Area Under ROC Curve (AUC) score of 88.5% on the Bird Audio Detection (BAD) challenge dataset. In the study by Cakir et al. [11], contributes in the domain of sound event detection, specifically targeting the intricate challenge of identifying multiple simultaneous sound events within audio recordings. It is done using CRNN constructed with convolutional layers for feature extraction and recurrent layers for temporal modeling, almost the same as [10]. Studies are also done to detect musical instrument from audio recordings using deep learning techniques [12]. They used CNNs, RNNs, and their hybrid variants, adept at learning representations from audio signals that aid in distinguishing between different musical instruments.

TABLE I. SUMMARY OF EXISTING TECHNIQUES.

Reference	Objective	Methodology	Detection Accuracy
[10]	Bird Audio Detection	Convolutional Recurrent Neural Network	88.5%
[11]	Detect Sound from a polyphonic event	Convolutional Recurrent Neural Network	70.0%
[12]	Instrument detection and audio content processing	Recurrent Neural Network - Long Short - Term Memory	84.0%
[13]	Understand CNN for audio detection	Convolutional Neural Network	40.0%
[14]	Investigate the benefits of doing a deep CNN	Deep Neural Network	92.0%
[15]	Birdsong Detection	Deep Neural Network	93.0%

Incze et al. [13] investigates the performance of CNN to recognize bird sound from audio files. The CNN model shows 80% accuracy for two bird classes, as expected. The performance decreases with more classes introduced. With 10 classes it is around 40% and at 50 classes accuracy is at 20%. Aggarwal et al. [14] demonstrates the performance of Deep Neural Network (DNN) to classify bird species from audio files. The model shows accuracy of 92%, and the closing remarks mentioned increasing the class balancing filter to increase accuracy. Disabato et al. [15] introduces ToucaNet, a deep neural network for bird song detection based on transfer learning, enabling faster training and improved accuracy. ToucaNet achieves detection accuracy comparable to state-of-the-art solutions in the literature, while significantly reducing computational complexity and memory demands. The proposed solutions demonstrate effectiveness and efficiency in bird song detection, making them suitable for at-scale intelligent data collection and analysis in the field using IoT devices.

B. Convolutional Neural Network

As elaborated earlier the model used in this paper is CNN. It is composed of convolutional blocks where each block is composed of a two-dimensional convolution filter varying levels of feature selectors depending on layer. A convolution filter will go over the entire image to find patterns of interest [16]. The filter is of two-fold, allows to use less parameters. Activation functions and optimizers change the stored values in the convolution filter rapidly and converge to values that decrease the overall loss of the model and therefore increase accuracy [17]. Then each layer of the network it is normalized with a mean of zero and a variance of one typically called batch normalization. The benefits of batch normalization are allowing the network to train faster, make some non-linear activation functions stay in operational regions. The next layer in the convolution block is the max pooling layer. By doing max pooling we are removing portions of the image that are not important in terms of features and we are only focusing on close matches that are picked up by the convolution layer [18]. Fully connected layers have a connection from the entry neuron layer to the output neural layer. The dropout layer helps to normalize the layers after the dense layer by excluding a random percentage of neurons for every training iteration. The final layer typically consists of SoftMax activation, which converts the network's output into probabilities across different classes like phonemes, words, etc [19]. Common optimization algorithms like stochastic gradient descent (SGD) or Adam are used for the purpose of minimizing differences between predictions and actual levels [20]. The choice of the loss function depends on the task.

III. METHODOLOGY

In this study, we have used three major stages, shown in Fig. 1, which include workable audio data collection, spectrogram extraction and training and validating CNN models.

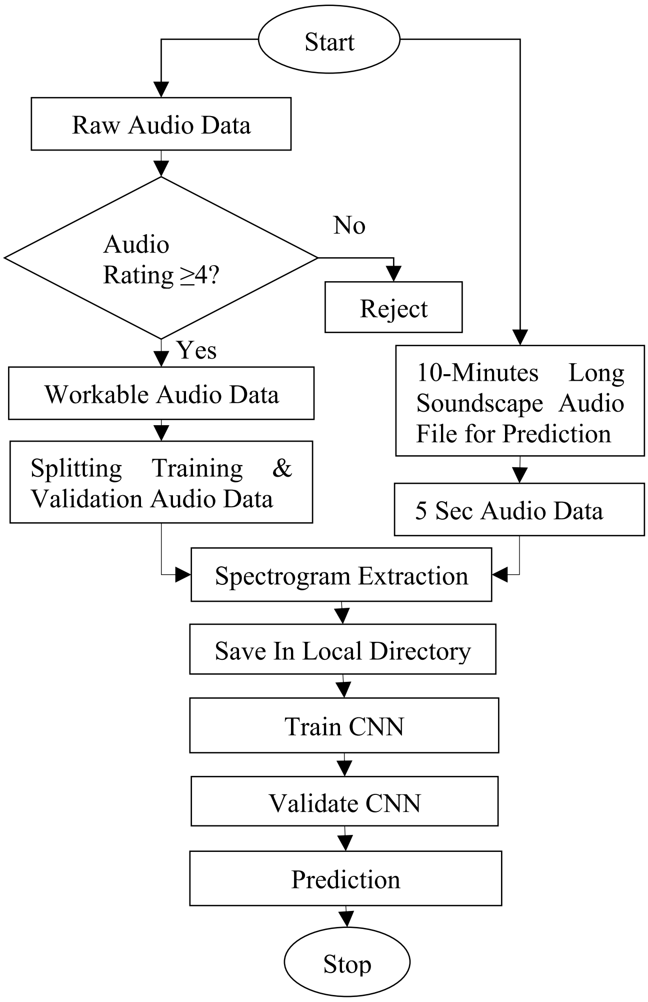

Fig. 1. Workflow Diagram of the Proposed Methodology.

Workable audio data are collected from the raw data based on the audio rating. If the audio rating is greater than or equal 4, then that audio file is considered. At the same time, 10- minutes long soundscape with artificial background noise to mimic an outside environment are collected for prediction.

That soundscape is converted into small 5 sec audio files. In the next stage, the audio file is converted into spectrogram and saved into local directory. In the last stage, the ML models are trained and validated using those spectrograms. Once the training and validation score is acceptable, the models are ready to predict with the prediction spectrograms.

IV. Experimental Details

A. *Technical Stack*

For the development process of this work, BeoShock (a high-performance computer system provided by Wichita state university) has been used. Since this is a computationally intensive job, BeoShock allows for a fast-acting execution. We used Jupyter Notebook in this high-performance platform for running the code. Setup for Jupyter requires a change of Python module from 3.2 to 3.10. Then we create a configuration file to allow BeoShock access to VScode for easy programming. Once this setup is processed, we change the kernel of the Jupyter notebook and begin with our dataset. The decision to implement this work on BeoShock is not only solely based on its computational ability but also on its scalability.

B. *Datasets*

The raw data has been collected from Kaggle [21]. The data contains a list of species and audio file. This data has many columns where the main three columns are primary_labels, rating, and filename. These columns allow us to make our dataset by creating a directory with the species where the filename is extracted from Xeno-canto only if the rating is of high quality. Within the dataset we end up identifying 40 species which all contain an audio rating greater than four on a scale of five. The intention for using high quality files is to prevent volatility between different files being run through the models training set. Low quality files would change the distribution and possibly impede on the convolution layers feature development. By focusing on files with ratings greater than four on a scale of five, we aim to maintain a consistent standard of data input. Fig. 2 shows the refinement of the dataset of how many audio files are there for each species that train the model. Once the audio files are extracted from Xentocanto's API and stored in our local host in BeoShock, the model is ready to be supervised in learning. These audio recordings are stored in another directory which has folders of the 40 different species.

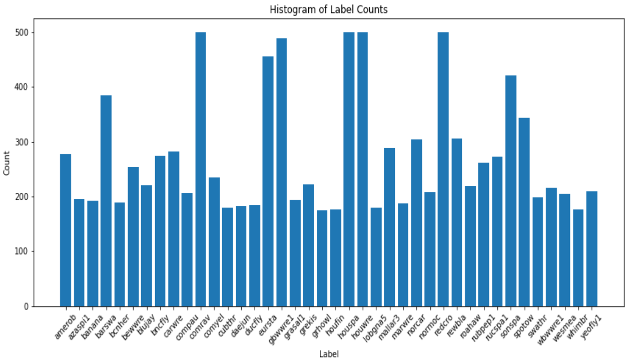

Fig. 2. Species count in the input dataset.

C. *Spectrograms*

From the sound clips, we are able to get the spectrograms of each audio file and save them in another working directory. These spectrograms, being visual representations of sound, serve as the input for our model's training. Each spectrogram encapsulates crucial frequency and time-based information essential for the model to pick distinct patterns among various bird species. Now we have five second spectrograms extracted from each audio file which are labeled for future reference. The library librosa has a simple function called feature.melspectrogram where it chunks the audio file being passed using a sample rate of 32,000 with the following hop length Equation (1).

$$Hop\ Length = \frac{Signal\ Length \times Sample\ Rate}{Spectrogram\ Shape - 1} \quad (1)$$

This gave us a window size that is uniform throughout our depiction of the spectrograms. The choice of a sampling rate of 32,000 and a specific hop length during the spectrogram extraction process is deliberate [22]. This selection impacts the granularity and resolution of the spectrograms, influencing the model's ability to capture fine details in the audio data. Fig. 3 shows the first 12 spectrograms of the training species.

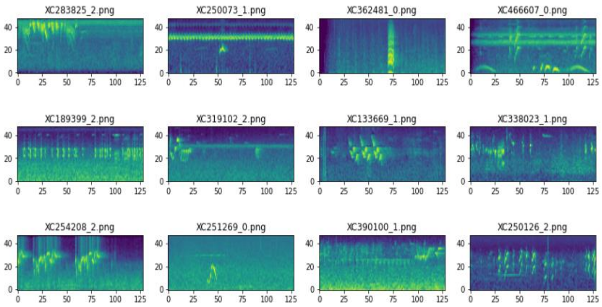

Fig. 3. First 12 spectrograms of training species.

By increasing sample rate, we are able to gain a greater frequency resolution. This is prevalent in the implementation for the Short Time Fourier Transform (STFT) because there is a budget between frequency and time resolution [23]. The STFT can be chosen to have a higher frequency resolution by selecting a window size of the time vector. The high frequency spectral components are removed by implementing a window function on the time vector. The window function tapers the ends of the sampled window to zero to attenuate any high frequency components [24]. The window function used in our CNN model is the Hann window function which is a standard smooth bell shape window function. The next parameter to consider is the overlap of the windows. Time resolutions are increased by increasing the amount of overlap between windows. If a window is 512 samples and an overlap is 511 then it is like taking a Discrete Fourier Transform (DFT) at each point on the time vector and doing 512 DFT's this would give the highest time resolution. The next step is the Discrete Cosine Transform (DCT). Before this transform resampling time-frequency spectrogram act as a low pass filter. However, since this low pass filtering in frequency domain is a high pass filter in the time domain. Once the spectrograms are extracted the model is ready for training.

D. *Convolutional Neural Network*

As elaborated earlier, the models used in the paper are two different sizes of CNN model. Each model is composed of four convolutional blocks where each block is composed of a two-dimensional convolution filter that are all 3 x 3 filters but have varying levels of feature selectors depending on layer and model. For both models the dimensions chosen is 2 x 2 for each convolution block. Both the models are trained with a batch size of 32, validation split of 20% and 50 epochs. The architecture similarity for both the models is chosen for a reason. If the architecture is kept different, then the two models would have a greater distribution variance of prediction, but the total sum of their predictions would yield inaccurate predictions. So, in practice we are keeping the distributions similar enough to be logical but unique enough to be more robust.

1) Model I

Model I is larger than Model II. The model contains four convolution blocks. The number of feature selectors increases by powers of two where the first layer contains 16 features then 32, 128 and then 256 for layer four. There is then a global average pooling to preserve spatial information of the model. Then a fully connected dense layer with 256 output connections and finally a 50% dropout layer to reduce overfitting after the dense layer.

2) Model II

Model II is like Model I in architecture construction but they differ in feature selector size. The difference comes from how many features are implemented in each layer. There are first 16 features then 32, 64 and finally 128 features on the fourth convolution block.

V. Results and Discussion

There are three independent results that are discussed to encompass the entire scope of work done in the ML models. First, the training and validation accuracy of both the models are described. Then run time is compared and at last, prediction results are elaborated.

A. Training and Validation Accuracy

Model I worked best with a greater number of feature selectors for our dataset producing 97.69% training accuracy and 71.92% validation accuracy. The training and validation accuracy of model II is 86.99% and 69.71% respectively, shown in Table II. The accuracy is lesser than that of model I but it is trivial to expect a model with more parameters would perform better if given the same architecture. Model II has a 116% faster run time than that of model I. So, if the project scales significantly and the models maintain similar accuracy metrics the decision to train model II over model I might be preferred.

TABLE II. ACCURACY AND RUN TIME COMPARISON BETWEEN TWO MODELS.

Parameter	Model I	Model II
Training Accuracy (%)	97.69	86.99
Validation Accuracy (%)	71.92	69.71
Completed Epoch	29.00	33.00
Total Run Time (mins)	51.10	59.30

B. Run Time Comparison

If the model doesn't improve in its accuracy between three consecutive epochs the training would discontinue and stay with the last metrics calculated, saving on computational demand. Model I stop early at epoch 29 after three consecutive attempts to make the model better as demonstrated in Fig. 4. Model II completes more epochs to reach optimal parameter values, shown in Fig. 5. The average epoch run time is 93 seconds and took 33 epochs to run; this equates to a run time of about 51 minutes and 10 seconds with 119,784 tunable parameters for the model. As stated earlier, model I completes its training in 29 epochs with an average run time of 123 seconds which equates to 59 minutes and 30 seconds of training time and having 414,760 tunable parameters. As stated earlier the decision on which model is "better", is questionable when considering training run time, computation requirements and overall accuracy.

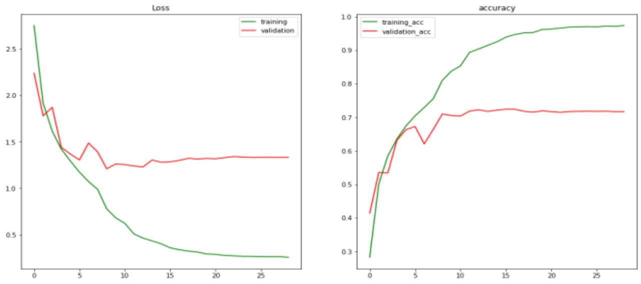

Fig. 4. Model I stopping epochs early to conclude maximized metrics.

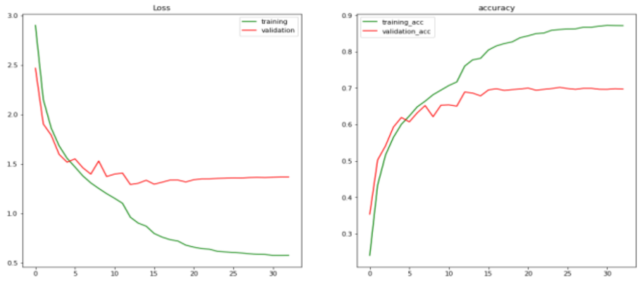

Fig. 5. Model II stopping epochs early to conclude maximized metrics.

C. Prediction From Soundscapes

The concern of predictions being accurate is considered in the over encompassing model by implementing soundscapes. The soundscape is a 10-minute audio file with artificial background noise to mimic an outside environment. There are also moments of silence that are observed throughout the soundscape. These 10-minute clips are processed into five second chunks and converted to spectrograms. There are multiple birds, no birds or one bird in any five second segment of the soundscape.

TABLE III. TRUNCATED EXAMPLE SOUNDSCAPE PREDICTION.

Row Id	Sec	Birds	Prediction
28933_SSW_45	45	sonspa	Nocall
28933_SSW_50	50	sonspa	sonspa
28933_SSW_55	55	sonspa	sonspa
28933_SSW_60	60	sonspa	sonspa
28933_SSW_65	65	Sonspa	Nocall
28933_SSW_70	70	sonspa	sonspa

By generating the soundscape artificially, we mimic the model's performance of a real-world test without having to physically record and produce new audio segments. This also assures us that we are properly predicting the bird species per every five second segment by having the soundscape intervals labeled beforehand. These labeled intervals are unseen from the model until prediction is processed. When processing a soundscape, the prediction adheres to the number of birds found in the 10-minute clip. In this paper, the soundscape '28933_SSW_20170408.ogg' outputs 89 birds out of 120 chunks of spectrograms which is 86.77% accurate. Table III shows a truncated example of what the soundscape will look like after it has been processed. The four main columns to understand are seconds, birds, and prediction. The second identifies the time the bird is found in the 10-minute audio clip, "birds" is the true label, prediction what our model presumes the bird species is. Each bird species is identified through aliases in which the name is shortened like sonspa is actually Song Sparrow.

VI. Conclusion

Traditional methods of avian song identifications face challenges due to the complex characteristics of avian vocalizations, which are characterized by a diversity of song types, regional variations, and individual nuances. In this work, we propose a method to identify real time avians and their respective song using two different sizes of CNN models. In the first phase, both the CNN models are trained and validated with the preprocessed audio files after converting into spectrograms over 40 bird species. In the second phase, the conclusive forecast is derived from a dataset comprising 10-minute soundscapes, which undergoes processing to generate 120 spectrograms at 5-second intervals. According to the experimental findings, the integrated model demonstrates an approximate 86% accuracy in predicting the overall count of bird species within each soundscape. The future of avian song identification holds exciting possibilities, ranging from improved species-specific recognition to real-time monitoring and broader ecological insights.

References

[1] M. T. Lopes, L. L. Gioppo, T. T. Higushi, C. A. A. Kaestner, C. N. Silla Jr. and A. L. Koerich, "Automatic Bird Species Identification for Large Number of Species," IEEE International Symposium on Multimedia, Dana Point, CA, USA, 2011, pp. 117-122, doi: 10.1109/ISM.2011.27.

[2] D. A. Nelson, "The Importance of Invariant and Distinctive Features in Species Recognition of Bird Song", The Condor, Volume 91, Issue 1, 1 February 1989, Pages 120–130, https://doi.org/10.2307/1368155.

[3] A. Marini, A. J. Turatti, A. S. Britto and A. L. Koerich, "Visual and acoustic identification of bird species," IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), South Brisbane, QLD, Australia, 2015, pp. 2309-2313, doi: 10.1109/ICASSP.2015.7178383.

[4] N. Priyadarshani, S. Marsland, and I. Castro, "Automated birdsong recognition in complex acoustic environments: a review", Journal of Avian Biology, 49: jav-01447. https://doi.org/10.1111/jav.01447

[5] R. Bardeli, D. Wolff, F. Kurth, M. Koch, K-H. Tauchert and K-H. Frommolt, "Detecting Bird Songs in a Complex Acoustic Environment and Application to Bioacoustic Monitoring", Patt. Recog. Letters, Vol.31, No.12, pp.1524-1534, 2010.

[6] T.S. Brandes, "Automated Sound Recording and Analysis Techniques for Bird Surveys and Conservation", Bird Cons. Int'l, Vol.18, pp.163-173, 2008.

[7] C.H. Lee, S.B. Hsu, J.L. Shi and C.H. Chou, "Continuous birdsong recognition using Gaussian mixture modeling of image shape features", IEEE Transactions on Multimedia, vol. 15, no. 2, pp. 454-464, 2012.

[8] S. -h. Zhang, Z. Zhao, Z. -y. Xu, K. Bellisario and B. C. Pijanowski, "Automatic Bird Vocalization Identification Based on Fusion of Spectral Pattern and Texture Features," IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Calgary, AB, Canada, 2018, pp. 271-275, doi: 10.1109/ICASSP.2018.8462156.

[9] Z. K. Abdul and A. K. Al-Talabani, "Mel Frequency Cepstral Coefficient and its Applications: A Review," IEEE Access, vol. 10, pp. 122136-122158, 2022, doi: 10.1109/ACCESS.2022.3223444.

[10] E. Cakir, S. Adavanne, G. Parascandolo, K. Drossos and T. Virtanen, "Convolutional recurrent neural networks for bird audio detection," IEEE Conference on European Signal Processing Conference (EUSIPCO), Kos, Greece, 2017, pp. 1744-1748, doi: 10.23919/EUSIPCO.2017.8081508.

[11] E. Çakır, G. Parascandolo, T. Heittola, H. Huttunen and T. Virtanen, "Convolutional Recurrent Neural Networks for Polyphonic Sound Event Detection," IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 25, no. 6, pp. 1291-1303, June 2017, doi: 10.1109/TASLP.2017.2690575.

[12] S. M. Elghamrawy and S. Edin Ibrahim, "Audio Signal Processing and Musical Instrument Detection using Deep Learning Techniques," IEEE International Japan-Africa Conference on Electronics, Communications, and Computations (JAC-ECC), Alexandria, Egypt, 2021, pp. 146-149, doi: 10.1109/JAC-ECC54461.2021.9691427.

[13] Á. Incze, H. -B. Jancsó, Z. Szilágyi, A. Farkas and C. Sulyok, "Bird Sound Recognition Using a Convolutional Neural Network," IEEE International Symposium on Intelligent Systems and Informatics (SISY), Subotica, Serbia, 2018, pp. 000295-000300, doi: 10.1109/SISY.2018.8524677.

[14] S. Aggarwal and S. Sehgal, "Classification of Bird Species using Audio processing and Deep Neural Network," IEEE International Conference on Intelligent Computing Instrumentation and Control Technologies (ICICICT), Kannur, India, 2022, pp. 138-143, doi: 10.1109/ICICICT54557.2022.9917735.

[15] S. Disabato, G. Canonaco, P. G. Flikkema, M. Roveri and C. Alippi, "Birdsong Detection at the Edge with Deep Learning," IEEE International Conference on Smart Computing (SMARTCOMP), Irvine, CA, USA, 2021, pp. 9-16, doi: 10.1109/SMARTCOMP52413.2021.00022.

[16] F. -I. Chou, Y. -K. Tsai, Y. -M. Chen, J. -T. Tsai and C. -C. Kuo, "Optimizing Parameters of Multi-Layer Convolutional Neural Network by Modeling and Optimization Method," IEEE Access, vol. 7, pp. 68316-68330, 2019, doi: 10.1109/ACCESS.2019.2918563.

[17] S. Albawi, T. A. Mohammed and S. Al-Zawi, "Understanding of a convolutional neural network," IEEE International Conference on Engineering and Technology (ICET), Antalya, Turkey, 2017, pp. 1-6, doi: 10.1109/ICEngTechnol.2017.8308186.

[18] T. Guo, J. Dong, H. Li and Y. Gao, "Simple convolutional neural network on image classification," IEEE International Conference on Big Data Analysis (ICBDA), Beijing, China, 2017, pp. 721-724, doi: 10.1109/ICBDA.2017.8078730.

[19] B. Ding, H. Qian and J. Zhou, "Activation functions and their characteristics in deep neural networks," IEEE Conference on Chinese Control And Decision Conference (CCDC), Shenyang, China, 2018, pp. 1836-1841, doi: 10.1109/CCDC.2018.8407425.

[20] E. M. Dogo, O. J. Afolabi, N. I. Nwulu, B. Twala and C. O. Aigbavboa, "A Comparative Analysis of Gradient Descent-Based Optimization Algorithms on Convolutional Neural Networks," IEEE International Conference on Computational Techniques, Electronics and Mechanical Systems (CTEMS), Belgaum, India, 2018, pp. 92-99, doi: 10.1109/CTEMS.2018.8769211.

[21] "BirdCLEF2021- Birdcall Identification - Identify bird calls in soundscape recording",www.kaggle.com/competitions/birdclef-2021/data.

[22] N. R. Koluguri, G. N. Meenakshi and P. K. Ghosh, "Spectrogram Enhancement Using Multiple Window Savitzky-Golay (MWSG) Filter for Robust Bird Sound Detection," IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 25, no. 6, pp. 1183-1192, June 2017, doi: 10.1109/TASLP.2017.2690562.

[23] Z. K. Abdul and A. K. Al-Talabani, "Mel Frequency Cepstral Coefficient and its Applications: A Review," IEEE Access, vol. 10, pp. 122136-122158, 2022, doi: 10.1109/ACCESS.2022.3223444.

[24] H. -K. Le, V. -P. Hoang, V. -S. Doan, M. -T. Hoang and N. P. Dao, "Performance Analysis of Convolutional Neural Networks with Different Window Functions for Automatic Modulation Classification," IEEE International Conference on Information and Communication Technology Convergence (ICTC), Jeju Island, Korea, Republic of, 2022, pp. 153-157, doi: 10.1109/ICTC55196.2022.9952750.