IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received 28 March 2024, accepted 18 April 2024, date of publication 23 April 2024, date of current version 1 May 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3392649

RESEARCH ARTICLE

Enhancing Gun Detection With Transfer Learning and YAMNet Audio Classification

N. HARIHARA VALLIAPPAN[1], SAGAR DHANRAJ PANDE[2], AND SURENDRA REDDY VINTA[1], (Member, IEEE)

[1]School of Computer Science and Engineering, VIT-AP University, Amaravati, Andhra Pradesh 522237, India
[2]School of Engineering and Technology, Pimpri Chinchwad University, Pune, Maharashtra 412106, India

Corresponding author: Surendra Reddy Vinta (vsurendra.cse@vitap.ac.in)

This work was supported by the Open Access funding provided by VIT-AP University.

ABSTRACT Identification of the type of gun used is essential in several fields, including forensics, the military, and defense. In this research, one of the powerful deep learning architectures is applied to identify several types of firearms based on their gunshot noises. For the purpose of extracting features from the audio data, the suggested technique makes use of YAMNet, an effective deep learning-based classification model. The Mel spectrograms created from the collected features are used for multi-class audio classification, which makes it possible to identify different types of guns. 1174 audio samples from 12 distinct weapons make up the study's extensive dataset, which offers a varied and representative collection for training and evaluation. We achieve a remarkable accuracy of 94.96% by employing the best hyperparameter changes and optimization methods. The findings of this study make a substantial contribution to the domains of forensics, military, and defense, where precise gun type identification is crucial. Applying deep learning and mel spectrograms to analyze gunshot audio demonstrates itself to be a promising strategy, providing quick and accurate categorization. This research emphasizes the effectiveness and relevance of using YAMNet, an AI-driven model, as a superior answer to the issues of real-world weapon detection.

INDEX TERMS Gun type identification, YAMNet, transfer learning, multi-class audio classification.

I. INTRODUCTION

Guns have always been a source of concern because of their potential for harm, especially when they come into the hands of the wrong people. The demand for stronger security measures have been further highlighted by the rise in terrorist incidents. The ability to recognize certain firearm types may considerably improve security and safety procedures in both civilian and military applications. For crime scene investigations and automatic recognition procedures, gun model identification systems with high recognition rates are needed.

The objective of this research work is to create a system for automatically identifying and categorizing various gun types. We use YAMNet, a pre-trained neural network well-known for its efficiency in feature extraction from audio data, to do this [1]. We can precisely extract discriminative features from gunshot audio by combining the strength of deep learning with the extensive feature representations provided by YAMNet.

YAMNet, a pre-trained neural network, uses the MobileNetV1 architecture, which includes depthwise-separable convolutions. YAMNet has been built to categorize audio signals into one of 521 unique categories from the AudioSet corpus, utilizing its capacity to analyze audio waveforms [11]. From the audio signal, frames are extracted and handled in batches by model. We use YAMNet, a state-of-the-art transfer learning method is renowned for its efficiency in feature extraction from audio data, to accomplish this goal. We can precisely extract discriminative features from gunshot sounds and enable effective classification of gun models by utilizing the strength of deep learning and the rich feature representations delivered by YAMNet. [2]

In this work, we make use of an 1174 audio sample collection that has been meticulously selected and is linked to one of 12 distinct gun types. We seek to achieve high

The associate editor coordinating the review of this manuscript and approving it for publication was Catherine Fang.

accuracy in recognizing and categorizing gun types based on their acoustic characteristics through rigorous training and evaluation methods.

By putting this gun model identification system into place, we intend to offer helpful resources for forensic analyses, pressing security circumstances, and military operations. The findings of this study can aid in developing security tactics, enhancing personal safety, and improving decision-making in reaction to occurrences involving firearms.

The contribution of this research is described as follows:

- The key contribution of this work is the use of YAMNet as a feature extractor, employing its pre-trained talents to extract rich and beneficial characteristics from gunshot audio.
- Research and evaluate several types of Keras layers to build our training model. In an effort to determine the ideal architecture for maximizing the performance of our gun model identification system, we examined a variety of layer layouts.
- To improve the precision of our model, pay special attention to hyperparameter optimization. By modifying variables like learning rate, batch size, and regularization techniques, we examined the best combination that improves accuracy and performance on unseen data.

The paper is structured as follows: Section II provides the relevant studies in gun model identification. Section III-A discusses the dataset. Section III-C details the proposed model, which includes the use of the YAMNet architecture. Section III-D provides a detailed description of the model. Section IV covers the findings and their consequences. Finally, Section V discusses future research avenues and the potential effects of our results.

By creating a strong gun model identification system, we want to improve security measures and offer insightful data for forensic analyses and military operations, thereby enhancing people's overall sense of safety.

II. RELATED WORKS

Numerous studies have been undertaken on the subject of gun model classification and gunshot recognition. To get accurate and trustworthy results, researchers have used a variety of procedures and techniques. This section outlines some noteworthy earlier efforts in this field.

Sengul Dogan used an H-tree pattern-based approach to classify different gun models in one significant study. They used the techniques for Support vector machines (SVM), K-nearest neighbors (KNN), and Neighborhood component analysis (NCA). They collected 2,130 audio samples from 28 distinct gun types for their dataset [4]. Another study by Rahul Nijhawan and Sharik Ali Ansari used Vision Transformers rather than conventional CNN models to examine the detection of firearms from gunshots. They made use of the 117 audio files that made up the UrbanSound collection. Their method showed the possibility of alternative deep learning architectures by achieving an accuracy of 93.87 percent [5]. In research by Junwoo Park and Youngwoo Cho, distinct audio samples from warfare settings were used to categorize gunshots in video games. They spoke about the assault stance and ultimately discovered the direction and dictate the gunfire. The BGG dataset, which had 2,195 samples, covered 37 distinct types of firearms. Their method has a 93.6 percent accuracy rate [6]. For the automated detection of gunshots, Ur Rahman, Sami, Khan, Adnan, Abbas, and Sohail presented a hybrid approach. The classifiers SVM, Tree, and KNN are used to distinguish between a gunshot and a regular scream. Their method included several strategies and had a 94.6 percent accuracy rate [7]. Based on Gaussian mixture models, Djeddou and Touhami created a feature selection technique for classifying weapons. With 230 bullets were fired from 14 firearms [8]. Additionally, Bajzik investigated convolutional neural networks (CNNs) for gun detection. They tested the system's effectiveness using loud gunshots [9].

Unlike these earlier studies, the focus of our research is on the use of transfer learning with YAMNet, a highly trained model that can handle noisy input. We use the mel spectrogram to express significant features. This innovative method seeks to improve the robustness and accuracy of systems used to classify gun models and identify gunshots. The investigations listed above have had a considerable impact on the categorization of gun models and the detection of gunshots. Further study will look at the efficacy of YAMNet, which is assisted by mel spectrogram representations, in the context of identifying gun models from gunshot sounds. This project aims to validate the performance and usability of these methodologies, therefore contributing to improvements in the area.

III. METHODOLOGY

A. DATASET DESCRIPTION

The dataset used in this study comes from two separate sources: the Gunshot Audio dataset [25] and the Gunshot Audio Forensics Dataset [26]. For training and assessment purposes, these datasets offer a wide-range and comprehensive collection of gunshot audio samples. The Gunshot Audio Dataset includes recordings of nine distinct gun types and was gathered through an open-access site. The audio recordings were gathered from films on YouTube that were made available to the public; thus, they included a range of gun noises. A total of 851 files, or numerous audio samples, were used to represent each gun model.Every audio file in the dataset was standardized to a duration of two seconds and a sampling rate of 44100 Hz. In addition, meticulous quality checks were carried out to make sure there were no unwanted sounds or disconnected audio parts [10].

The Gunshot Audio Forensics Dataset is an extensively assembled collection of distinct data files. These recordings, which were made as part of the NIJ Grant 2016-DN-BX-0183 study, were made in a rural Arizona community during the summer of 2017. Three distinct firearm types are represented

TABLE 1. Comparison of previous models for gunshot detection.

Author	Year	Method	Methodology	Dataset	Positives	Negatives
Sengul Dogan [4]	2021	H-tree pattern	SVM, kNN	YouTube videos	Robust results with cross-validation	Low accuracy with minimal datasets
Nijhawan Rahul et al. [5]	2022	Transformer learning	Transformer encoder	TRECVID, UrbanSound	High training accuracy	Binary classification, no model type specified
Park Junwoo et al. [6]	2022	Deep neural layers	CNNs, Transformers	BGG dataset	Comparison of 5 advanced models	No hyperparameter tuning
Jiri Prinosi et al. [7]	2022	Residual networks	SVM, DNN	UrbanSound8k, Free Firearm Sound Library	Improved feature extraction with residual convolution	Computationally expensive
Mustapha Djeddou et al. [8]	2013	Decision trees	GMM, SVM	Self-recorded data	Hierarchical classification framework	Accuracy varies between gun models
Jakub Bajzik et al.[9]	2020	Machine learning frameworks	VGG16, ResNet18	UrbanSound8K	Testing with various parameters	Low training epochs
T. Tuncer et al. [10]	2021	Finger pattern feature	IRF feature selector, KNN	Kaggle dataset	Effective feature selection with IRF	High time consumption
Alberto Tena et al.[12]	2022	YAMNet deep network	Autoencoders, YAMNet	COVID-19 datasets	Improved accuracy with YAMNet preprocessing	Limited hyperparameter tuning
Shin, Mikyong Deborah [23]	2023	Deep neural networks	MCC-based YAMNet	TUT-SED, ESC-50, SINS datasets	Detailed YAMNet explanation	No model comparison
Nicolini et al.[18]	2023	Transfer learning	YAMNet with hyperparameter tuning	Riboni et al. dataset	Efficient and scalable method	Small dataset

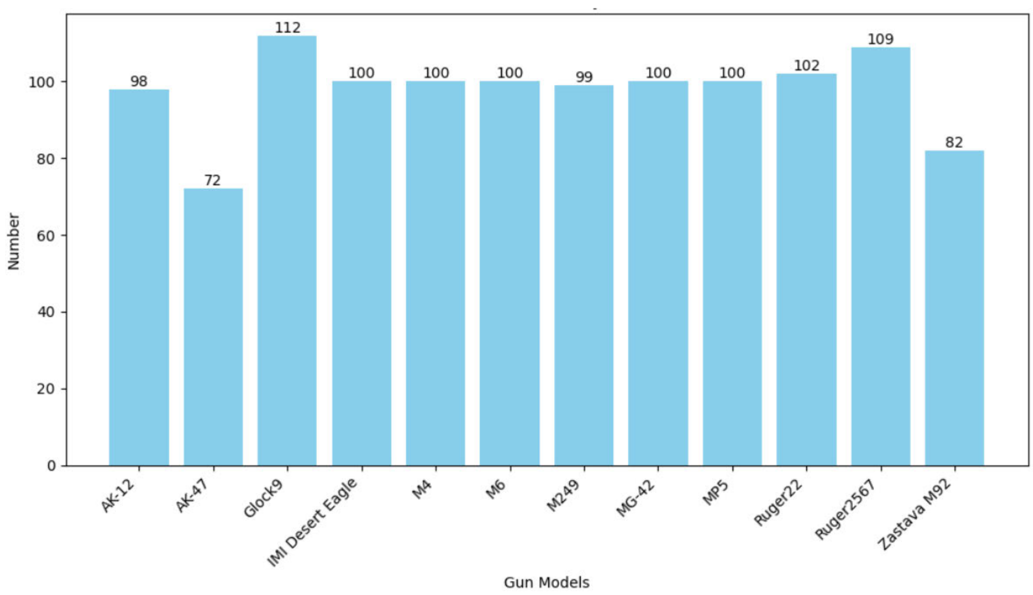

FIGURE 1. Number of audios with respective gun.

by audio recordings in the dataset: the Glock 19 (111), Ruger Blackhawk (102), and Ruger 10/22 (109). Twenty separate recording locations were used to capture each handgun, giving researchers a thorough insight into the acoustic traits and variances unique to each gun model.The total number of audio files used to train our model are 1174 (Figure 1).

With a total of twelve different gun types represented, the combination of these two datasets results in a rich and varied collection of gunshot audio samples. This large dataset is a significant resource for training and assessing our gun detection technology. With the use of transfer learning, we use YAMNet audio classification models to improve the precision and effectiveness of gun detection systems.

B. DATA PREPROCESSING

The WAV file format, which is a standard for audio data, is utilized for the audio samples in this study report. To ensure compatibility and ideal input for the YAMNet audio classification model, numerous procedures are taken during the preprocessing stage.

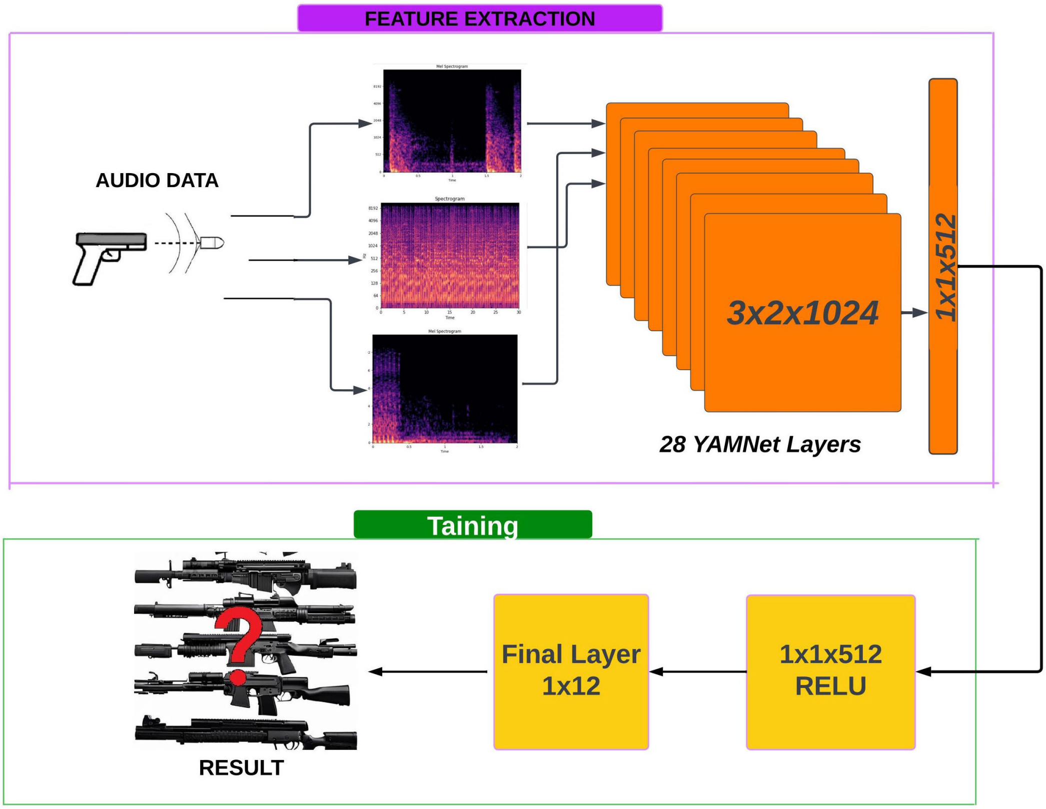

FIGURE 2. Work flow a model.

First, a float-tensor representation of the WAV files is created once they have been loaded. The audio data may be processed and analyzed numerically as a result of this conversion. Additionally, the audio samples are resampled to a sampling rate of 16 kHz in order to achieve uniformity and standardization. The audio signals are normalized and uniformized using the resampling process. Additionally, the mel spectrogram representation is used to extract significant characteristics from the audio. The frequency-domain visual representation of the audio signals is provided by the mel spectrogram [13]. It is frequently employed in jobs involving audio processing and captures the spectral properties of the audio.

C. PROPOSED MODEL

The suggested model for classifying guns using a mix of deep learning and audio processing methods is shown in Figure 2. Beginning with a depiction of a gun with ammo releasing sound waves, the illustration shows how our technique unfolds sequentially. The sound waves are transformed into a Mel spectrogram—an effective audio feature representation—using TensorFlow's audio decoding capabilities. Following that, the YAMNet model is used as a feature extractor to efficiently get high-level data from the Mel spectrogram [13]. The Keras library is used to create a succession of fully connected layers from the YAMNet outputs. Finally, based on the observed features, these layers determine the gun's identity. The mechanics of how this identification is accomplished will be addressed in the subsequent paragraphs.

MobileNets use a simplified design with depthwise separable convolutions to build lightweight deep neural networks, which serve as the foundation for YAMNet's efficient audio categorization capabilities. This design includes two global hyperparameters that efficiently balance the trade-off between latency and accuracy, allowing model builders to customize the model size based on a unique application needs and limitations. MobileNets, as the model's backbone, allow it to analyze and identify audio inputs efficiently and accurately [24].

Using YAMNet as a feature extractor and transfer learning as a learning tool, we propose a novel technique for categorizing weapons. We have between 72 and 112 audio samples for each type of gun in our library. To offer resilience and avoid overfitting during the training and validation phases, we adopt k-fold cross-validation. Using k-fold cross-validation, the dataset is divided into k equal-sized folds, with the remaining k-1 folds are being used for training. Each fold serves as the validation set once. This process is repeated k times, and the average performance over all folds is calculated to get the final accuracy [5], [14].

We use the Sparse Categorical Cross entropy loss function, which works well for multi-class classification applications like classifying guns, to train the model. When each sample is a member of a single class, this loss function effectively handles the situation. The difference between the genuine

Algorithm 1 Gun Classification using YAMNet and Neural Network

```
Audio Dataset Preprocessing:
    Load audio samples with associated labels.
    Convert samples to mono and resample to 16 kHz.
    Apply regular expressions for noise reduction and
feature extraction.
YAMNet-based Feature Extraction:
    Initialize YAMNet, leveraging pre-trained models.
    Apply regular expressions for precise feature
extraction.
Dataset Segmentation and Model Training:
    Segment dataset into training, validation, and test
subsets. for each training epoch do
each batch in training set
    Utilize YAMNet for audio sample batch processing
and feature extraction.


Construction of Neural Network Model:
    Construct neural network architecture using
tf.keras.Sequential:
        Input Layer: Input shape defined by extracted
feature dimensions.
        Hidden Layer: Dense layer with activation
functions.
        Output Layer: Dense layer to classify into firearm
classes.
Model Compilation and Training:
    Compile and train neural network model using
extracted embeddings.
    Assess and validate model accuracy using validation
and test embeddings.
```

class label and the anticipated probability distribution is measured. Sparse categorical cross entropy has the following formula: [15]

$$L(\text{Sparse Categorical Crossentropy}) = -\sum_{i=1}^{N}(Y_{tv} \cdot \log(Y_{pv})) \tag{1}$$

where Y_{tv} represents the actual likelihood that the sample belongs to class i. Y_{pv} represents the predicted likelihood that the sample belongs to class i.

The Adamax optimization method, a variation of the popular Adam optimizer, is employed in the model that we suggest. Particularly, the infinite norm serves as the foundation for Adamax, a first-order gradient-based optimization technique. This approach of optimization was chosen because of its exceptional capacity to dynamically modify the learning rate depending on the underlying properties of the data.

The Adamax method makes use of both Adaptive Moment Estimation (Adam) and the infinite norm principles, making it suitable for learning time-variant processes, such as those seen in voice data when noise conditions change over time. Adamax can handle scenarios where gradients might differ greatly and diverge in particular directions successfully by integrating the infinity norm. Given that the fluctuations in the data may be quite dynamic and diverse, it is a fascinating alternative for challenging audio data processing jobs [16].

$$m = \beta_1 \times m + (1 - \beta_1) \times \text{gr} \tag{2}$$

Exponentially weighted infinity norm,

$$W_v = \max(\beta_2 \times W_v, |\text{gr}|) \tag{3}$$

Weight update,

$$\text{weight} = \text{weight} - L_r \times \frac{m}{(W_v + \epsilon)} \tag{4}$$

where, gr is a gradient of he weight,β_1 and β_2 are decay rate for first and second momentum, and L_r is a learning rate [16]. The first moment estimations, which serve as the moving average of previous gradients, have an exponential decay rate that is controlled by the β_1 parameter, which has an initial level of 0.9. It functions as a momentum term that enables the optimizer to recall prior gradients and modify the update direction as necessary. A more responsive optimizer can be produced with a smaller value of β_1, which causes a quicker decline of the previous gradients. On the other hand, a higher value of β_1 makes the optimizer more steadfast in its course and might result in smoother convergence.

We extend the robust YAMNet feature extractor and expand its capabilities in our proposed model for classifying guns by adding Keras Dense layers for further processing. The feature vector is sent through a Dense layer with 512 nodes that has been activated using the Rectified Linear Unit (ReLU) activation function after pertinent features have been extracted from the Mel spectrogram using YAMNet. By introducing non-linearity and resolving the vanishing gradient issue, the ReLU activation improves our model's ability to learn complicated patterns and representations [17].

$$f(value_{\text{updated}}) = \max(0, value_{\text{initial}}) \tag{5}$$

L2 regularisation is featured as a strong approach to improve the neural network's generalisation performance. In order to prevent overfitting, L2 regularizationl is sometimes referred to as weight decay, penalizes the magnitudes of the model's weights. L2 regularisation promotes the model to choose smaller, more evenly distributed weights by deterring big weight values and introducing a regularization component to the loss function. This regularization method helps reduce the danger of overfitting and improve generalization to new data. It is particularly useful when the model is complicated or when the training data is few. λ is a regularization parameter and L(W) is a loss function of W. The following is the L2 formula [21], [22]:

$$L(W) = \text{Loss}(V_{\text{true}} - V_{\text{pred}}) + \lambda \times \|w\|^2 \tag{6}$$

We add a Dropout layer after the initial Dense layer to combat overfitting and enhance generalisation. The Dropout layer lowers interdependent learning among neurons and increases model resilience by randomly setting a portion of the input units to 0 during training [21].

Following the Dropout layer, we deploy another Dense layer with the number of nodes matching the entire number of classes in our gun classification issue. We use softmax activation function on the output layer to obtain the desired categorization. The softmax function converts the real-valued output vector into a probability distribution in which the total of the probabilities across all classes equals 1. This makes it possible for us to interpret the output as class probabilities, making it possible to understand the model's predictions in a way that is simpler and more logical. The following mathematical equation may be used to represent the softmax function, where Au is the input vector and e is the exponential function [22]:

$$\sigma(Au)_i = \frac{e^{Au_i}}{\sum_{k=1}^{L} e^{Au_k}} \tag{7}$$

The softmax function turns each element in the input vector Au represented by Au_i into a value between 0 and 1 in this formula. A correct probability distribution is provided by the normalization element in the denominator, which guarantees that the output values add to 1.

The model is trained across a large number of epochs, allowing it to learn from the training data repeatedly. The addition of thick layers after YAMNet's feature extraction enhances the model's ability to distinguish intricate patterns in the data and generate accurate predictions for gun classification. Our proposed model integrates ReLU, Dropout, and softmax functions, as well as the adaptability of Keras Dense layers, feature extraction capabilities of YAMNet, and feature extraction capabilities of YAMNet to achieve state-of-the-art accuracy in gun classification while fostering model stability and generalization.

D. MODEL DESCRIPTION

We load the WAV files and convert them to float tensors before resampling the audio to a single channel at a sampling rate of 16 kHz (Mel spectrogram) to preprocess the audio data. Then, using the Mel spectrograms, we extract features using YAMNet to create a 1024-dimensional embedding. The total number of parameters used to train YAMNet is 3.7 million [23].

We use mel spectrograms, a well-liked preprocessing approach for acoustic deep learning systems, to get the audio data ready for YAMNet. The audio signal's short-time Fourier transform (STFT), which transforms the signal from the time domain to the frequency domain, is computed first before creating the log spectrogram. Following that, the STFT's magnitude is determined, and the logarithm of the magnitude is computed to produce the log spectrogram. This modification improves the representation of characteristics for audio analysis tasks and aids in compressing the spectrogram's dynamic range [3], [27].

TABLE 2. YAMNet architecture.

Name	Type	Input size	Filter size
Conv1	-	48x48x48	3x3x3
Conv2	Depthwise	48x48x48	3x3x3
Conv2	Pointwise	48x32x64	1x1x32x64
conv3	Depthwise	24x16x64	3x3x64
conv3	Pointwise	24x16x128	3x3x128
conv4	Depthwise	24x16x128	3x3x128
Conv4	Pointwise	24x16x128	1x1x128x128
Conv5	Depthwise	12x8x128	3x3x128
Conv5	Pointwise	12x8x256	1x1x128x256
Conv6	Depthwise	12x8x256	3x3x256
Conv6	Pointwise	12x8x256	1x1x256x256
Conv7	Depthwise	6x4x512	3xx3x256
Conv7	Pointwise	6x4x512	1x1x256x512
Conv8	Depthwise	6x4x512	3x3x512
Conv8	Pointwise	6x4x512	1x1x512
Conv9	Depthwise	6x4x512	3x3x512
Conv9	Pointwise	6x4x512	1x1x512
Conv10	Depthwise	6x4x512	3x3x512
Conv10	Pointwise	6x4x512	1x1x512
Conv11	Depthwise	6x4x512	3x3x512
Conv11	Pointwise	6x4x512	1x1x512
Conv12	Depthwise	6x4x512	3x3x512
Conv12	Pointwise	6x4x512	1x1x512
Conv13	Depthwise	3x2x1024	3x3x512
Conv13	Pointwise	3x2x1024	1x1x512x1024
Conv14	Depthwise	3x2x1024	3x3x1024
Conv14	Depthwise	3x2x1024	1x1x1024x1024
Pool1	Average pooling	1x1x1024	3x2
FC1	Fully connected	1x1x512	1024x512
classifier	softmax	1x1x512	-

$$Freq_{\text{mel-S}} = 2595 \cdot \log(1 + \frac{Freq}{700\,\text{Hz}}) \tag{8}$$

The relationship between frequency and mel-spectrogram frequency is seen in the following above equation. Where,$Freq_{mel-S}$ is Mel frequency, $Freq$ is a linear frequency.

Then, utilizing transfer learning and YAMNet audio classification, our research attempts to enhance gun detection. Based on the AudioSet-YouTube corpus, the pre-trained deep neural network YAMNet predicts 521 audio event types. The architecture used by the model, known as MobileNetV1 (depthwise-separable convolution), was trained using more than 632 audio events taken from YouTube videos. [12] A 1D convolution layer with a kernel size of 3 × 3 is the first layer in the YAMNet design, which is followed by multiple layers with increasing filter sizes up to 1024. To avoid overfitting, a global average pooling (AP) layer is included. Fully connected (FC) layers, measuring 1024 and 64 pixels, and a softmax layer to anticipate the kind of audio event round out the model [3], [15].

Each input channel (feature map) is convolved independently with its own set of filters during a depth-wise convolution.

By adding more layers to the architecture, we further create our model. Our unique model has a dense layer with 512 units, an input layer with 1024 dimensions, and a ReLU activation function to increase non-linearity and solve the vanishing gradient problem. In order to create the final forecast, which represents the various sorts of guns, we use a second thick layer with 9 units. We use the Adam optimizer and sparse categorical cross-entropy loss to train our model. To avoid overfitting, early halting with patience of 3 is used.

In this study, we provide a model that achieves precise and effective classification and shows the effectiveness of employing YAMNet for gun type recognition. Rich auditory characteristics may be obtained by using mel spectrograms as input pre-processing, which makes our method a viable choice for applications requiring the identification of guns despite a shortage of data.

IV. RESULT

In this study, we used YAMNet, a potent audio classification model, to improve gun detection. 1174 audio samples of 12 different types of guns were used in our research, and they were gathered from a variety of internet sources (Kaggle [25] and Cadre Forensics [26]). Each audio's mel-spectrogram served as the input for the YAMNet model's feature extraction process.

To improve the model's performance, several trials with different epochs and learning rates have to be done. To avoid overfitting and restoring the ideal weights, we implemented an early-stopping callback. After properly shuffling the dataset to guarantee a randomized distribution, the training, validation, and test datasets were divided into the following proportions: 80:10:10. The tests were carried out on a system with 8 GB of RAM and an Intel i5 quad-core CPU from the 10th generation clocked at 1.00 GHz.

On our gun classification model, we inspected the performance of several optimizers and examined their associated accuracies. The optimizers Adam, RMSprop, Adagrad, SGD, Adamax, Adadelta, and FTRL were among those put to the test. After comprehensive testing, we found that RMSprop, with an accuracy of 93.42%, was closely behind Adamax, which had the greatest accuracy of 94.60%. Adam had a respectable performance, obtaining an accuracy of 93.14%. Adagrad, SGD, Adadelta, and FTRL, on the other hand, had significantly lower accuracies (Table 3).

These results show that our gun classification model performs best with Adamax optimizers, whereas Adadelta and FTRL works less well.

In addition to the previously shown findings, we also examined the effectiveness of our gun classification model using several loss functions, including Sparse Categorical Crossentropy, Kullback-Leibler, and Squared-Hinge. Table 4 lists the acquired accuracy, F1 scores, and precision scores for each loss function:

As seen in the table, the Kullback-Leibler and Squared-Hinge loss functions had much lower accuracy ratings than the SparseCategoricalCrossentropy loss function, which had a maximum accuracy of 93.14%.

In comparison to Kullback-Leibler and Squared-Hinge loss functions, the SparseCategoricalCrossentropy loss function, which is the most efficient, assures greater accuracy, F1 score, and precision. This implies that the best outcomes for audio-based classification tasks, such as gun classification, rely greatly on the choice of an appropriate loss function.

TABLE 3. Optimizer relutls.

Optimizer	Accuracy	F_1 Score	Precision	Recall	Specificity
Adam	93.14%	92.69%	93.80%	91.20%	94.50%
RMSprop	93.42%	92.57%	92.79%	92.00%	94.00%
Adagrad	85.49%	82.22%	84.25%	81.00%	87.20%
SGD	91.87%	90.14%	89.08%	91.50%	90.20%
Adamax	**94.60%**	**93.49%**	**93.40%**	**95.10%**	**94.00%**
Adadelta	71.82%	62.14%	55.67%	73.50%	72.00%
FTRL	75.02%	60.72%	51.71%	76.30%	74.20%

TABLE 4. Comparison of different loss functions.

Loss Function	Accuracy	F_1 Score	Precision
SparseCategoricalCrossentropy	**93.14%**	**92.69%**	**93.80%**
Kullback-Leibler	14.55%	12.61%	12.46%
Squared-Hinge	15.55%	11.78%	11.46%

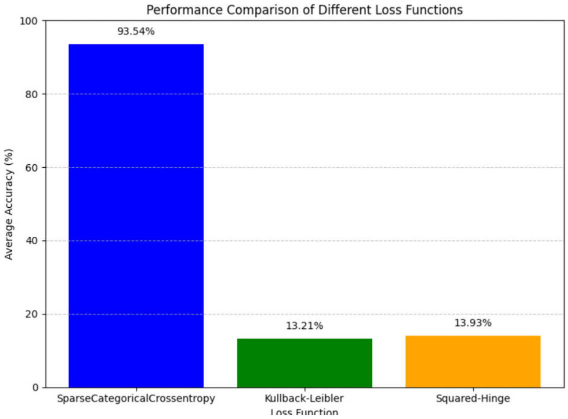

FIGURE 3. Visual representation of different loss functions.

TABLE 5. Epochs table.

Number of Epochs	Accuracy	F1 Score	Recall
50	82.06	0.80	0.79
100	86.14	0.84	0.82
200	89.33	0.87	0.86
300	92.12	0.90	0.89
400	**94.33**	0.93	0.92
500	93.98	0.93	0.92

In the epochs section, we investigated the effects of changing the number of epochs on the effectiveness of our model for classifying guns. Intriguing conclusions came from our study. With 50 epochs as a starting point, we found an accuracy of 82.06%. The accuracy rose to 86.14% as the epoch count approached 100. Surprisingly, it took 400 epochs to get the best accuracy of 94.33%. But when the epochs were multiplied by 500, the accuracy fell to 93.96%. These findings suggest that while accuracy initially increases with more epochs, there is an ideal limit beyond which additional improvements are minimal.

We reviewed the effects of various cross-fold values on the precision of our gun classification model as part of

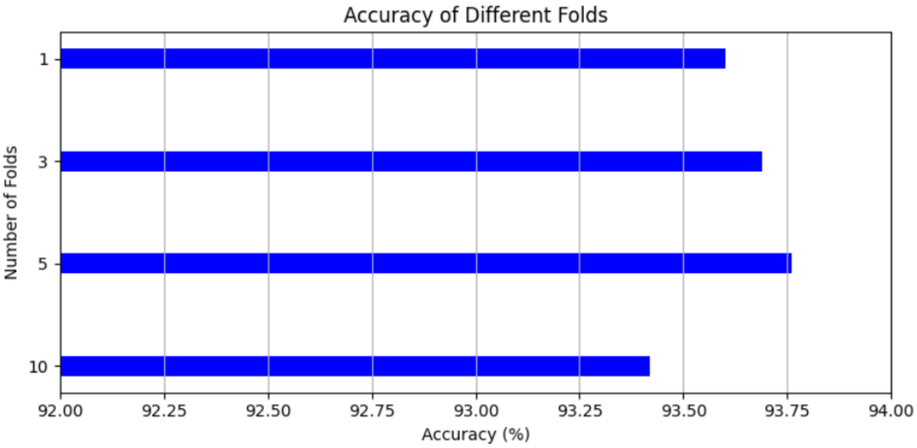

FIGURE 4. Accuracy of different folds.

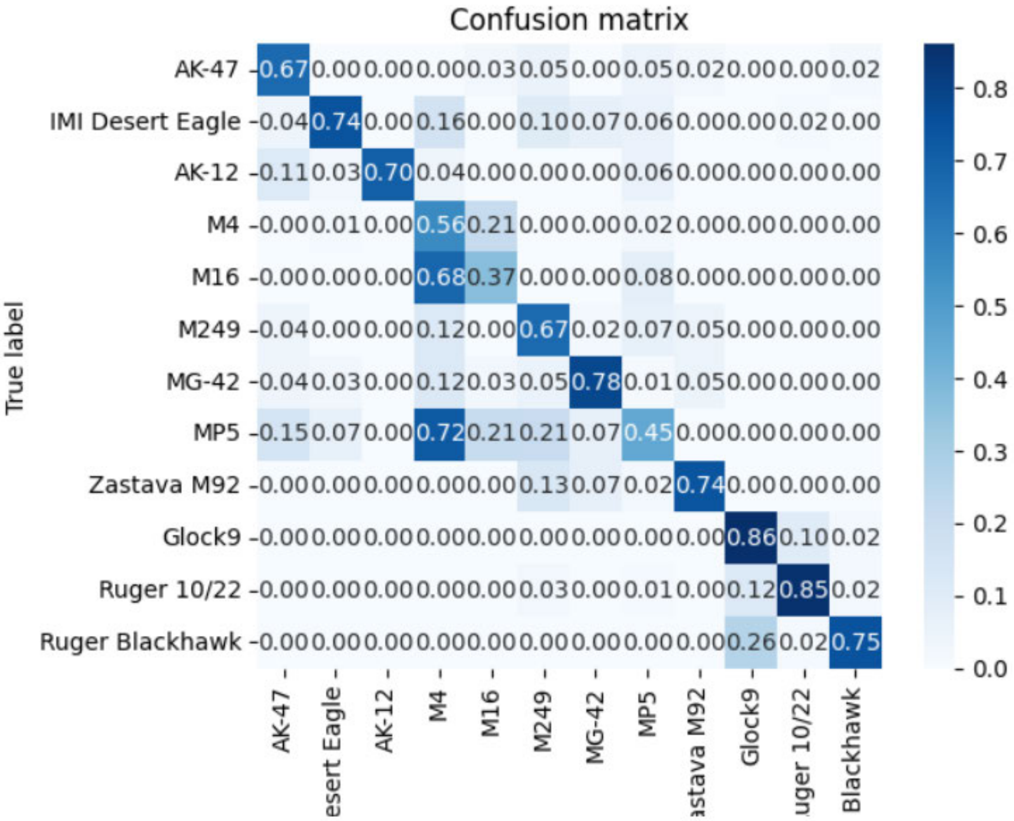

FIGURE 5. Model's confusion matrix for gun categorising.

the cross-fold validation process. We performed several trials with settings of 10, 5, 3, and 1. We discovered that, interestingly, 5 folds had the best accuracy (93.76%), closely followed by 3 folds (93.69%). The accuracy was marginally lower in both 10 and 1-fold settings, at 93.42% and 93.60%, respectively. These findings indicate that a reasonable number of folds, such as 5, could assist with model performance by providing a rigorous yet manageable validation procedure, as shown in Figure 4.

TABLE 6. Hybrid models with metrics.

Models	Accuracy (%)	F1 Score (%)	Recall (%)
Fully connected layers	94.96	94	94.8
SVM	93.6	93.66	93.9
KNN (k=1)	90.06	91.3	91
KNN (k=3)	90.14	90.12	90.43
KNN (k=5)	89.32	89.99	89.34

After YAMNet feature extraction, our study moved into the training stage, using a deep learning methodology using

TABLE 7. Comparison of different methods and proposed model.

Study	Year	Dataset	Methods Used	Accuracy (%)
[10]	2021	851 gunshots for 8 gun models	IRF with KNN classifier	94.48
[5]	2022	117 audios for 3 classes	Transformer Learning with resnet50	93.87
[6]	2022	2195 gunshots for 37 gun models	Transformer based models	88.27
[7]	2022	-	SVM and neural networks (MFCC)	87.0
Proposed model	2023	1173 audios with 12 classes	YAMNet	**94.96**

Keras layers. Notably, the results were remarkably better than those of conventional SVM and KNN algorithms (with k = 1, 3, and 5). The accuracy improvements attained using the Keras layers demonstrated the effectiveness of utilizing deep learning techniques. This striking difference in accuracy emphasizes our model's better predictive power, highlighting its capability to identify complex patterns and produce exact classifications, and reaffirms the superiority of our method to categorizing guns. The accuracy is shown in Table 6.

An informative assessment of the model's classification performance may be found in the final confusion matrix. The values in the matrix show the level of accuracy of each class of firearms tested at. It is striking that classes like Glock 9, and Ruger 10/22 have high accuracies of 0.86, and 0.85 respectively, demonstrating the model's competence in accurately recognizing these weapons. Classes like M4, M16, and MP5, on the other hand, have lower accuracy levels, indicating possible areas for development. Overall, the confusion matrix is an essential tool for evaluating how well the model categorizes firearms. Figure 5 shows confusion matrix.

In conclusion, our suggested model for YAMNet-based gun classification has shown excellent results when combined with the optimized hyperparameters. We trained the model for 400 epochs using the Adamax optimizer, with a learning rate of 0.001, L2 regularization of 0.001, and a dropout rate of 0.2. Sparse Categorical Crossentropy was the preferred loss function. Through careful experimentation, We were able to get a fine accuracy of **94.96%**,F1 score of **94.40%**and a Precision of **94.13%** These results demonstrate how well our algorithm performs in reliably recognizing gun models from acoustic data. The overall performance of the model was much enhanced by the hyperparameter fine-tuning. Our results show that YAMNet and the selected hyperparameters are appropriate for a robust and accurate gun classification, which contributes significantly to the field of audio-based classification systems.

V. CONCLUSION

Our study thoroughly assesses gun model recognition using YAMNet via transfer learning and selective hyperparameter changes. By carefully training the proposed model with a wide range of parameter values, we show that using YAMNet with fine-tuned hyperparameters outperforms existing deep learning and hybrid techniques in this area. An important component of our research is the investigation and integration of extra Keras layers to improve the model's training efficacy. Despite the dataset's restrictions, our model achieves impressive accuracy over a wide range of gun types, demonstrating the validity of our technique. While accepting the dataset limitations, our findings are encouraging, pointing to future research paths such as dataset enlargement for increased model accuracy and the development of novel strategies to handle environmental variables.

In conclusion, this work demonstrates the potential of YAMNet-based transfer learning in weapon model identification, which has significant benefits for forensic and military applications. Our findings, particularly the rigorous tuning and testing of each parameter presented in the results section, demonstrate our method's ability to set a new standard in the area.

VI. FUTURE SCOPE

In terms of future potential, our study opens the door to more sophisticated investigation by including generative adversarial networks (GANs) into our model. By adding GANs, we may improve the realism of the synthesised data and perhaps get over the constraints imposed by our current dataset. This approach could make it possible to get beyond obstacles like a lack of training data and raise the model's accuracy for various types of guns. Another fascinating direction is to modify the model to account for differences brought on by shifting environmental factors. We anticipate a promising trajectory towards an even more accurate gun model identification by utilizing cutting-edge methodologies and embracing a larger dataset, hence enhancing the usefulness of our methodology in forensic and military situations. The potential of our strategy might be further increased by investigating transfer learning with other previously trained models or by experimenting with ensemble methodologies.

REFERENCES

[1] E. Tsalera, A. Papadakis, and M. Samarakou, "Comparison of pre-trained CNNs for audio classification using transfer learning," *J. Sensor Actuator Netw.*, vol. 10, no. 4, p. 72, Dec. 2021.

[2] S. Patil and K. Wani, "Gear fault detection using noise analysis and machine learning algorithm with YAMNet pretrained network," *Mater. Today, Proc.*, vol. 72, pp. 1322–1327, 2023.

[3] W. Chen, H. Kamachi, A. Yokokubo, and G. Lopez, "Bone conduction eating activity detection based on YAMNet transfer learning and LSTM networks," in *Proc. 15th Int. Joint Conf. Biomed. Eng. Syst. Technol.*, 2022.

[4] S. Dogan, "A new fractal H-tree pattern based gun model identification method using gunshot audios," *Appl. Acoust.*, vol. 177, Jun. 2021, Art. no. 107916.

[5] R. Nijhawan, S. A. Ansari, S. Kumar, F. Alassery, and S. M. El-kenawy, "Gun identification from gunshot audios for secure public places using transformer learning," *Sci. Rep.*, vol. 12, no. 1, pp. 1–5, Aug. 2022.

[6] J. Park, "Enemy spotted: In-game gun sound dataset for gunshot classification and localization," in *Proc. IEEE Conf. Games (CoG)*, 2022, pp. 56–63.

[7] J. Bajzik et al., "Independent channel residual convolutional network for gunshot detection," *Int. J. Adv. Comput. Sci. Appl.*, vol. 13, no. 4, pp. 950–958, 2022.

[8] M. Djeddou and T. Touhami, "Classification and modeling of acoustic gunshot signatures," *Arabian J. Sci. Eng.*, vol. 38, no. 12, pp. 3399–3406, Dec. 2013, doi: 10.1007/s13369-013-0655-5.

[9] J. Bajzik, J. Prinosil, and D. Koniar, "Gunshot detection using convolutional neural networks," in *Proc. 24th Int. Conf. Electron.*, Lithuania, 2020, pp. 1–5, doi: 10.1109/IEEECONF49502.2020.9141621.

[10] T. Tuncer, S. Dogan, E. Akbal, and E. Aydemir, "An automated gunshot audio classification method based on finger pattern feature generator and iterative relieff feature selector," *Adıyaman Üniversitesi Mühendislik Bilimleri Dergisi*, vol. 8, no. 14, pp. 225–243, 2021.

[11] L. G. Martins. (Mar. 2, 2021). *Transfer Learning for Audio Data With YAMNet*. TensorFlow Blog. [Online]. Available: https://medium.com/analytics-vidhya/understanding-the-mel-spectrogram-fca2afa2ce53

[12] A. Tena, F. Clarià, and F. Solsona, "Automated detection of COVID-19 cough," *Biomed. Signal Process. Control*, vol. 71, Jan. 2022, Art. no. 103175, doi: 10.1016/j.bspc.2021.103175.

[13] A. Patel, S. Degadwala, and D. Vyas, "Lung respiratory audio prediction using transfer learning models," in *Proc. 6th Int. Conf. I-SMAC (IoT Social, Mobile, Analytics Cloud) (I-SMAC)*, Dharan, Nepal, Nov. 2022, pp. 1107–1114.

[14] R. Baliram Singh, H. Zhuang, and J. K. Pawani, "Data collection, modeling, and classification for gunshot and gunshot-like audio events: A case study," *Sensors*, vol. 21, no. 21, p. 7320, Nov. 2021.

[15] A. K. Sharma, G. Aggarwal, S. Bhardwaj, P. Chakrabarti, T. Chakrabarti, J. H. Abawajy, S. Bhattacharyya, R. Mishra, A. Das, and H. Mahdin, "Classification of Indian classical music with time-series matching deep learning approach," *IEEE Access*, vol. 9, pp. 102041–102052, 2021.

[16] N. A. M. Ariff and A. R. Ismail, "Study of Adam and adamax optimizers on AlexNet architecture for voice biometric authentication system," in *Proc. 17th Int. Conf. Ubiquitous Inf. Manage. Commun. (IMCOM)*, Jan. 2023, pp. 1–4.

[17] H. Ide and T. Kurita, "Improvement of learning for CNN with ReLU activation by sparse regularization," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, May 2017, pp. 2684–2691.

[18] M. Nicolini, F. Simonetta, and S. Ntalampiras, "Lightweight audio-based human activity classification using transfer learning," in *Proc. 12th Int. Conf. Pattern Recognit. Appl. Methods*. ScitePress, 2023, pp. 783–789.

[19] M. R. K. Mookiah, R. Puch-Solis, and N. Nic Daeid, "Identification of bullets fired from air guns using machine and deep learning methods," *Forensic Sci. Int.*, vol. 349, Aug. 2023, Art. no. 111734.

[20] X. Ni, L. Fang, and H. Huttunen, "Adaptive l2 regularization in person re-identification," in *Proc. 25th Int. Conf. Pattern Recognit. (ICPR)*, Jan. 2021, pp. 9601–9607.

[21] E. Phaisangittisagul, "An analysis of the regularization between l2 and dropout in single hidden layer neural network," in *Proc. 7th Int. Conf. Intell. Syst., Model. Simul. (ISMS)*, Thailand, Jan. 2016, pp. 174–179, doi: 10.1109/ISMS.2016.14.

[22] M. Wang, S. Lu, D. Zhu, J. Lin, and Z. Wang, "A high-speed and low-complexity architecture for softmax function in deep learning," in *Proc. IEEE Asia–Pacific Conf. Circuits Syst. (APCCAS)*, Oct. 2018, pp. 223–226.

[23] M. D. Shin, "Adaptation of pre-trained deep neural networks for sound event detection facilitating smart homecare," *J. Abbreviated*. [Online]. Available: https://urn.fi/URN:NBN:fi:tuni-202305316375

[24] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "MobileNets: Efficient convolutional neural networks for mobile vision applications," 2017, *arXiv:1704.04861*.

[25] E. Aydemir. (Jun. 22, 2021). *Gunshot Audio Dataset*. [Dataset]. Kaggle. [Online]. Available: https://www.kaggle.com/datasets/emrahaydemr/gunshot-audio-dataset

[26] R. Lilien and J. Housma. (2019). *Gunshot Audio Forensic*. Cadre Research. [Online]. Available: http://cadreforensics.com/audio/

[27] (Mar. 6, 2020). *Understanding the Mel Spectrogram*. Analytics Vidhya. [Online]. Available: https://medium.com/analytics-vidhya/understanding-the-mel-spectrogram-fca2afa2ce53

N. HARIHARA VALLIAPPAN received the B.Tech. degree in computer science from Vellore Institute of Technology, Andhra Pradesh, with a focus on different technical and research-focused projects. He has published one patent on the defense sector along with many journals and conferences during his graduation. Beyond the classroom, he regularly engages in research programs across the world that are relevant to his area of interest, including AI/ML and data analytics.

SAGAR DHANRAJ PANDE received the Ph.D. degree in computer science and engineering from Lovely Professional University, Phagwara, Punjab, India, in 2021. He is currently an Assistant Professor with the School of Engineering and Technology, Pimpri Chinchwad University, Pune, Maharashtra, India. He has published and presented more than 60 articles in Springer, Elsevier, CRC, Taylor & Francis, and other reputable journals, which are Scopus-indexed and peer-reviewed journals. Also, he has published papers at international conferences springer on the topics of data mining, network security, the IoT, and its application. He has supervised several postgraduate students in cybersecurity, computer networks, communication, and the IoT. He is responsible for teaching artificial intelligence, deep learning, machine learning, cybercrime and security, and Python programming courses to undergraduate and postgraduate students. His research interests include deep learning, machine learning, network attacks, cyber security, and the Internet of Medical Things (IoMT). He received the Young Researcher Award and the Best Ph.D. Thesis Award from Universal Innovators, in 2022. Also, he received the Emerging Scientist Award from VDGOOD Professional Association, in 2021. He is also sharing his knowledge through his YouTube channel named sdpguruji https://www.youtube.com/c/SDPGuruji/playlists.

SURENDRA REDDY VINTA (Member, IEEE) received the Ph.D. degree from VBSP University, India, under the supervision of Prof. (Dr.) Sourab Paul and co-supervised by Dr. Raju Bukya. He is currently an Associate Professor with the School of Computer Science and Engineering, VIT-AP University, Amaravati, Andhra Pradesh, India. Under his guidance, four students are working for Ph.D. degree. A seasoned academician having more than 15 years of experience, he has published four books, such as *Programming in C*, *Programming in C++*, *Machine Learning*, and *Deep Learning* (I. K. International Publishing House Pvt. Ltd.) He has undergone industrial training programs during which he was involved in live projects with companies in the areas of SAP, railway traffic management systems, and visual vehicle counter and classification (used in the Metro rail network design). A prolific writer, he has published twenty patents and authored many research articles in web of science indexed journals. Additionally, he has presented research papers at many conferences in the areas of image processing, machine learning, deep learning, NLP, computer vision, features extraction, and programming, such as digital image processing, feature extraction, machine learning, deep learning, NLP, computer vision, C, Python, data structure, C++, and Java. He has professional memberships in IEEE, ISTE, and IET.

Open Access funding provided by 'VIT-AP University' within the CRUI CARE Agreement