IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 12 September 2023, accepted 17 September 2023, date of publication 22 September 2023,
date of current version 4 October 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3318015

SURVEY

A Survey of Audio Classification Using Deep Learning

KHALID ZAMAN[1], **MELIKE SAH**[2], **CEM DIREKOGLU**[3], **AND MASASHI UNOKI**[1], **(Member, IEEE)**
[1]Graduate School of Advanced Science and Technology, Japan Advanced Institute of Science and Technology, Nomi, Ishikawa 923–1292, Japan
[2]Computer Engineering Department, Cyprus International University, 99258 Nicosia, North Cyprus, Turkey
[3]Electrical and Electronics Engineering Department, Middle East Technical University, Northern Cyprus Campus, 99738 Kalkanli, Guzelyurt, Turkey

Corresponding author: Khalid Zaman (zaman.khalid@jaist.ac.jp)

This work was supported in part by the Grant-in-Aid for Transformative Research Areas (A) under Grant 23H04344, and in part by the Grant-in-Aid for Challenging Research (Exploratory) under Grant 23K18491.

ABSTRACT Deep learning can be used for audio signal classification in a variety of ways. It can be used to detect and classify various types of audio signals such as speech, music, and environmental sounds. Deep learning models are able to learn complex patterns of audio signals and can be trained on large datasets to achieve high accuracy. To employ deep learning for audio signal classification, the audio signal must first be represented in a suitable form. This can be done using signal representation techniques such as using spectrograms, Mel-frequency Cepstral coefficients, linear predictive coding, and wavelet decomposition. Once the audio signal is represented in a suitable form, it can then be fed into a deep learning model. Various deep learning models can be utilized for audio classification. We provide an extensive survey of current deep learning models used for a variety of audio classification tasks. In particular, we focus on works published under five different deep neural network architectures, namely Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), Autoencoders, Transformers and Hybrid Models (hybrid deep learning models and hybrid deep learning models with traditional classifiers). CNNs can be used to classify audio signals into different categories such as speech, music, and environmental sounds. They can also be used for speech recognition, speaker identification, and emotion recognition. RNNs are widely used for audio classification and audio segmentation. RNN models can capture temporal patterns of audio signals and be used to classify audio segments into different categories. Another approach is to use autoencoders for learning the features of audio signals and then classifying the signals into different categories. Transformers are also well-suited for audio classification. In particular, temporal and frequency features can be extracted to identify the characteristics of the audio signals. Finally, hybrid models for audio classification either combine various deep learning architectures (i.e. CNN-RNN) or combine deep learning models with traditional machine learning techniques (i.e. CNN-Support Vector Machine). These hybrid models take advantage of the strengths of different architectures while avoiding their weaknesses. Existing literature under different categories of deep learning are summarized and compared in detail.

INDEX TERMS Audio, speech, music, emotion, noise, classification, recognition, deep learning, CNNs, RNNs, autoencoders, transformers, hybrid models.

I. INTRODUCTION

Audio classification is the analysis and classification of audio signals into various categories. It is an important tool in audio signal processing as it helps to organize, analyze, and understand audio signals. By classifying audio signals, it is possible to better understand the underlying signal, its structure, and content, which can be useful for a variety of applications. There are various audio classification applications such as virtual assistants, automated voice translators, environmental sound classification applications, music genre identification, and text to speech. In general, audio classification can be broadly grouped into acoustic data classification, speech classification, music classification,

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

environmental sound classification and natural language classification [1].

Audio classification technology can help make everyday life easier by improving the accuracy of voice recognition technology used for hands-free device control. This can make it easier for people with disabilities or limited mobility to control devices in their home, enabling them to do things like turn on lights, adjust the thermostat, and open the door without having to physically move. It can also help improve the accuracy of voice-enabled digital assistants used in smart homes, cars, and other technology, enabling users to access the information they need quickly and easily. Moreover, audio classification can enable everyday life more convenient by allowing people to identify sounds quickly and accurately. For example, a smartphone application can use audio classification to recognize when a baby is crying and alert the parents. Audio classification technology can also be used to quickly recognize different types of music, enabling easy music streaming. In addition, audio classification can be used for security purposes, such as detecting intruders in a home or business place, and for automated customer service, such as providing automated responses to customer inquiries.

To achieve successful audio classification, the first step is to obtain annotated audio data. The model should be trained with annotated data to learn how to recognize and categorize different sounds. Despite the progress made in audio classification, it is still challenging to teach a machine the nuances of sound and classify it accordingly.

Traditional audio classification methods such as Support Vector Machines (SVM), K-Nearest Neighbors (KNN), Artificial Neural Networks (ANNs) and hidden Markov models (HMMs), have been used since the early 2000s. These methods are used to extract features from audio recordings and then use the features to classify the audio into different classes. An SVM is a powerful supervised machine learning algorithm that uses hyperplanes to separate data points into categories for classifying audio recordings with high accuracy [2], [3], [4], [5], [6], [7], [8], [9] KNN is a supervised learning algorithm for classification. It is used to find the nearest neighbors of an input instance and then uses the class of most of the neighbors to classify it. It is suitable for audio classification as it can use the features of the audio signal to accurately classify the signal [11], [12], [13], [14], [15]. ANNs are computational models based on the structure and functions of biological neural networks. They can be used for audio classification by learning the characteristics of audio samples and then using these characteristics to classify the samples into different classes [16], [17], [18], [19], [20]. Logistic regression is used to classify data into two classes for audio classification tasks such as speech recognition and music genre classification. In audio classification, it takes an audio signal as input, extracts features from it, and then classifies it into the desired output class [21]. The Naive Bayes classifier is a probabilistic classifier that is used to calculate the probabilities of each class for a given data point and then assigning the class with the highest probability. It is also relatively fast and simple to implement, making it a common choice for audio classification [22], [23], [24], [25]. HMMs are also commonly used to classify audio data. HMMs are particularly well-suited for audio classification because they can learn the underlying structure of the audio data and model the temporal dynamics of the audio signal [26], [27], [28], [29], [30]. A Gaussian Mixture Model, is a type of probabilistic model for audio classification with which each data point is assumed a mixture of different Gaussian distributions, which are then used to classify the audio test input data [11], [31], [32].

Deep learning is becoming increasingly popular for speech classification. With its ability to learn complex patterns, it can achieve better accuracy than traditional approaches. Traditional approaches usually divide audio classification into two processes: feature extraction and classification [33]. In the feature extraction process, relevant features are extracted from the audio data, then these features are used in the classification process to identify the audio data. However, deep learning models require large audio datasets for training the network and learning the features of each class automatically. Once trained, the model can be used to classify new audio samples.

Deep learning models perform better than traditional audio classification models. However, deep learning models for audio classification are capable of automatically extracting the high-dimensional features of samples from a large-scale dataset without manual feature extraction, as long as the input data contain all the relevant information of the original data [34]. Deep learning models are capable of achieving higher accuracy rates than traditional models. This is due to their ability to learn complex patterns and recognize subtle differences in audio data. Most deep learning models can learn faster and more accurately than traditional models, which makes them ideal for real-time audio classification and analysis. Obviously, deep learning models are more reliable when trained with a large number of samples to learn the task-specific features than the traditional machine learning-based method, especially for transformer-based methods, without local inductive bias in CNN. This is especially useful when dealing with audio data, which can be expensive and time consuming to label. Despite this, deep learning models can adapt to new data without requiring major modifications to the model architecture. Overall, deep learning is a powerful and flexible approach to audio classification since it is capable of surpassing traditional approaches in terms of accuracy, performance, reliability, scalability, and adaptability.

There are also some drawbacks of deep learning. Deep learning models require significant computing resources, including powerful graphics processing units and a large amount of memory, to train, which can be expensive and time consuming. Moreover, training and evaluation of audio classification systems with deep neural networks (DNNs) is only possible with a large amount of audio data; without a large dataset, the system can be unsuccessful [35]. Over-fitting occurs when a model has been over-trained on the data it has already acquired, leading to poor performance with new unseen data [36]. On the other hand, under-fitting may occur if a model has not been trained enough.

The deep learning models are heavily dependent on the quality of the data they are trained on. If the data are noisy, biased and incomplete, the model's results can be significantly impacted [37]. The utilization of deep learning models necessitates the handling of large amounts of data, raising concerns about data privacy and security. If data are misused by unscrupulous individuals, it can lead to serious repercussions, such as identity theft, financial loss and an infringement of one's privacy [38]. Therefore, deep learning models for audio classification have become widespread for their potential to solve complex tasks. However, there are some limitations such as high computational cost, lack of interpretability, over-fitting, data privacy and security concerns, lack of domain expertise, dependence on data quality, and unforeseen datasets [35], [36], [37], [38], [39].

The application of audio classification using deep learning has become increasingly common in various domains such as computer vision, natural language processing (NLP), healthcare, and industrial signal processing. This success has extended to speech recognition and music recommendation tasks. Deep learning models can also be used to detect anomalies in audio signals, such as background noise or other unwanted sounds. The need for automated sound classification systems is growing as their importance in our daily lives cannot be underestimated. Such systems are used in a wide range of areas, such as surveillance, voice assistance, chatbots, smart safety devices, and various real-world environments, including engineering, industrial, domestic, urban, road, and natural.

We provide an extensive survey of current deep learning models that are applied to a variety of audio classification tasks such as speech, noise, music, emotion, environmental sound, and acoustic scene. In particular, for audio classification, we review literature under five different deep learning architectures such as Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), autoencoders, transformers and hybrid models (hybrid deep learning models, such as CNN-RNN and hybrid deep learning models with traditional classifiers such as CNN-SVM). For each type of architecture, detailed information about the architecture components is also given. In addition, the existing literature in each deep learning architecture is summarized and compared in detail. Moreover, we also briefly discuss different audio datasets for classification tasks. Although there have been surveys on traditional audio classification methods [41], [42], [43], [44] and deep learning-based methods [40] [45], [46], [47], [48], [49] for specific audio classification task, none compare in detail to the literature on deep learning models with different deep learning architectures with varying applications, which is the contribution of this work. Corrêa et al. [44] provides an overview of the key methods for classifying music genres and exploring the symbolic representation of musical data. Fu et al. [43] also provides a survey of audio features and traditional classification methods for music classification. Dandashi and AlJaam [42] provide a review of audio processing and classification methods on the basis of acoustic events, genres, and scenes, as well as combinations of them. Reshma and Rajasree [41] briefly discuss different feature extraction techniques and classification algorithms for speech emotion recognition. Other surveys were conducted for deep learning models. Akinpelu and Viriri [40] provide a survey of approaches for classification of emotions from speech. Bansal and Garg [45] discuss research on environmental sound classification, covering topics such as pre-processing, feature extraction, and classification techniques. Bhangale and Kothandaraman [46] provide a survey of deep learning models for speech processing. They divide the literature into different learning groups; supervised, unsupervised, semi-supervised, and reinforcement learning. They also discuss applications in these domains. Roger et al. [47] outline the deep learning models for speech processing tasks. Abeßer [48] reviews deep learning-based methods for acoustic scene classification. Khan et al. [49] conducted a survey of CNN architectures for image and video classification, and speech recognition.

Previous surveys either focus on traditional classification methods [41], [42], [43], [44] or outline the deep learning models in a particular field, such as emotion recognition [40], [41], acoustic scene/event classification [42], [48], environmental sound classification [45] and speech processing [46], [47], [49]. Different from these surveys, we provide a broad discussion of audio classification using deep learning models. In addition, we divide the literature into different deep-learning architectures; CNN, RNN, autoencoders, transformers and hybrid models. For each architecture type, the main components of the deep learning architectures were explained and then methods that use these deep learning architectures were discussed and compared in detail. On the basis of the literature, we analyzed application areas for each deep learning architecture and explain commonly used audio datasets. Under each deep learning architecture, we compared the methods in terms of input data type, deep learning model, used dataset(s), performance and application area. In this way, more insights can be learned such as preferred input data types for different deep learning models, which deep learning models are more suitable for various audio classification tasks, suitable datasets for a particular task, and performance analysis. Furthermore, we outline future directions of deep learning in audio classification. We believe that this makes our survey stand out from other numerous surveys on audio classification.

To summarize, comparing our survey to other related surveys in the field, several distinct advantages and limitations come to light. Our survey paper distinguishes itself through its detailed exploration of deep learning architectures for audio classification, encompassing modern deep learning models and discusses application areas. While coverage of various deep learning models for audio classification, including hybrid models, makes it a valuable resource for those seeking insights into different architectures. In terms of limitations, our survey covering a broad spectrum of models on audio classification might limit the depth of analysis. Some other surveys focus on a specific domain ([40], [41], [42], [45], [46], [47], [48], [49] to conduct more comprehensive examinations.

Overall, our survey summarizes the current trends in audio classification using deep learning and provides future directions. We believe that it will help readers and researchers in this context.

The rest of the paper is organized as follows. Section II provides information on different deep learning architectures used in audio classification, Section III discusses the available audio datasets for classification and section IV elaborates on the discussion of all deep learning models.

II. AUDIO CLASSIFICATION USING DEEP LEARNING

The history of deep learning for audio classification dates back to the late 1990s. In 1998, Yann LeCun, at AT&T Bell Labs, proposed the use of CNNs for recognizing speech [1]. Since then, various audio applications have used CNNs such as for speech recognition, audio classification and music recommendation systems. In 2004, Sukittanon et al. [50] presented a method for speech detection using convolutional networks and a network architecture in the detection process that captures both long-term and short-term temporal and spectral correlations of speech. In 2006, Graves et al. [51] introduced a new and general method for temporal classification using RNNs in the speech recognition domain. In 2012, Abdel-Hamid et al. [52] published the first paper on speech recognition using CNNs. In the past decade, CNNs became increasingly common for audio classification, with research focusing on improving the accuracy and speed of these models. Researchers have applied various deep learning models such as RNNs, long short-term memory (LSTM) networks, autoencoders, transformers and hybrid models that combine different deep learning models to a variety audio classification tasks. These models have produced impressive results in various audio classification tasks.

In the following sub-sections, we categorize the literature in accordance with different deep learning architectures. We observed that CNNs, RNNs, autoencoders, transformers and hybrid models are commonly used for audio classification (Figure 1). Therefore, deep learning-based methods are explained under these categories. After investigation of audio classification methods, we observed that some deep learning methods use a single deep learning architecture such as CNN only, RNN only, autoencoder only or transformer only. In the review, methods that use a single deep learning architecture are explained under the specific architecture type. On the other hand, many deep learning methods combine various deep learning architectures to produce hybrid deep learning models. For example, many RNN methods initially use CNN for feature extraction, and then output of the CNN is given to the RNN. Similarly, autoencoders can be combined with RNNs (i.e. LSTM) or CNNs. Therefore, we considered such methods as hybrid deep learning models. Finally, a standalone deep learning model can be combined with traditional classifiers such as SVM, KNN or HMMs. For example, feature extraction can be achieved by a CNN model, and then classification is performed by using a traditional classifier like SVM. We consider these deep learning models with traditional classifiers as hybrid models as well. According to this categorization, methods are reviewed under these deep learning architectures. We first explain the architectural properties of these different deep learning architectures and then summarize related work for each category.

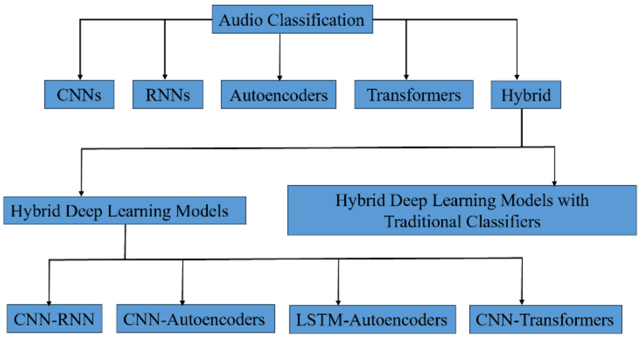


FIGURE 1. Classification of related work on different deep learning architectures.

The most commonly used evaluation criteria for all classification methods, including deep learning, are evaluation metrics evaluation. Metrics are a set of measures used to evaluate the performance of deep learning models to determine how well a model can learn from training data and to identify areas of improvement to optimize the model's performance. The most common of these metrics are accuracy, precision, recall, F1-score, ROC curve (receiver operating characteristic), and AUC (area under the ROC curve) score. Other metrics, such as confusion matrix, sensitivity, and specificity, may also be used to evaluate the performance of deep learning models [53], [54], [55].

Accuracy is the percentage of correctly classified data points by an algorithm compared with all data points, as shown by the following equation:

$$\text{Accuracy} = \frac{\text{Number of correct predictions}}{\text{Total number of predictions}} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} \ + \text{FN}}, \tag{1}$$

where TP, TN, FP, and FN denote true positive, true negative, false positive, and false negative, respectively. Precision is the ratio of correctly predicted data points to the total predicted data points and defined as:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{2}$$

Recall and sensitivity measure the proportion of correctly classified data points belonging to a particular class out of all data points classified belonging to that particular class in the dataset and calculated as:

$$\text{Recall/Sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{3}$$

Specificity is a metric for measuring the accuracy of a classifier in correctly identifying all the TNs in the dataset and can be calculated as:

$$\text{Specificity} = \frac{\text{TN}}{\text{TN} + \text{FP}} \tag{4}$$

F1-score measures the overall performance of a model by taking the harmonic mean of precision and recall and calculated as:

$$\text{F1-score} = \frac{2\,(\text{Precision} * \text{Recall})}{\text{Precision} + \text{Recall}} = \frac{\text{TP}}{\text{TP} + 1/2(\text{FP} + \text{FN})} \tag{5}$$

Area under curve (AUC) and Receiver operating characteristic curve (ROC), on the other hand are graphs that illustrate TP and FP rates for a given classification model. These graphs are used to evaluate the performance of a model regarding its ability to correctly classify true positives and true negatives.

A. CNN-BASED METHODS

In this section, we discuss CNN-based methods for audio classification.

1) CNN ARCHITECTURES AND INPUTS TO CNN

CNNs are commonly used for speech recognition, audio classification, music recommendation, audio source separation (i.e. separating different audio sources from a single recording), and many more application areas. CNNs have revolutionized the field of audio classification and enabled a wide range of applications. Inputs to CNNs are audio/speech signals; however, CNN-based methods usually do not use the raw one-dimensional (1D) signals as input. As a pre-processing stage, 1D audio/speech signals are converted from 1D signal to 2D signal. The 2D representation of the audio signal is then input to a CNN model. This 1D to 2D conversion is generally executed to generate spectrograms. Spectrograms capture spectrum frequencies of an audio signal since the audio signal varies with time [56], [57], [82]. Fast Fourier Transform (FFT), short-term Fourier Transform (STFT), Mel-Frequency, log-Mel-frequency, wavelet transform, and many other types of spectrograms can be used to convert 1D audio signals to a 2D representation, as shown in Figure 2. Discrete Fourier Transform (DFT) is a technique that is used for extracting features from raw audio signals. Particularly, it converts signals from the time domain to the frequency domain to acquire the phase and magnitude of every frequency component. However, DFT is not optimized and difficult to apply to real-time discrete signals. FFT, however, is an optimized application of DFT that can be applied to real-time discrete signals and defined as:

$$S(k) = \sum\nolimits_{n=0}^{N-1} S\,(\text{n})\, e^{-j\frac{2\pi}{N}kn} \tag{6}$$

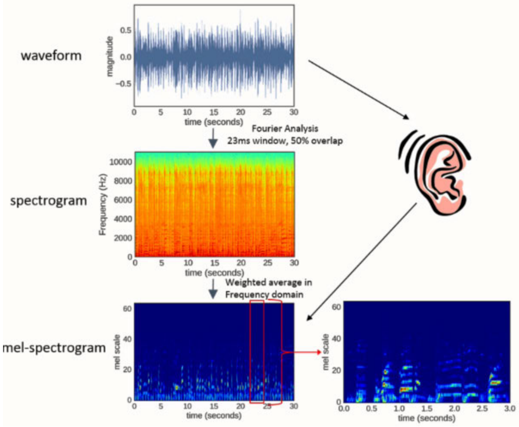


FIGURE 2. Sample spectrogram representation of an audio signal [69].

The magnitude spectrum, $|S(k)|$ of a signal is the magnitude of its frequency bin or frequency component number (k) at a given sample number (n). It is usually a complex value [82].

STFT is an improved form of Fourier Transform (FT) to provide temporal details of signals in both the time and frequency domains. Using a window in STFT, the signal is divided into fixed-sized time-domain segments. Each segment is then subjected to FT to reveal different features of the signal. In essence, STFT uses equally spaced, identical, and symmetrical bandpass filters in the frequency domain to analyze the signal. The mathematical formulation of any signal s(t), can be written as:

$$S(f, t) = \int_{-T}^{T} s\,(\tau)\, w\,(\tau - \text{t})\, e^{-j2\pi f\tau} \mathrm{d}\tau \tag{7}$$

To obtain this representation, signal s (t) is partitioned into segments, over a windowing function w (t) in Equation 7. The window length must be the same as the length of the signal segments, and it is assumed that the signal does not change (stationary) inside a window duration. The spectrogram is obtained by using STFT by taking the magnitude squared value of the time-frequency representation [82], [157], expressed as

$$\text{Spectrogram} = |\text{S(f, t)}|^2 \tag{8}$$

A Mel-spectrogram, however, can be obtained using the raw signal. Using the human auditory system, the Mel scale provides a linear scale in relation to Hertz using the following equation [55], [161].

$$M(f) = 2595\text{log}10(1 + \frac{f}{700}) \tag{9}$$

where $M(f)$ is the Mel frequency for a given frequency (f). It is derived from a logarithmic scale and related to the way humans perceive sound [161]. The formula is based on the assumption that frequencies have a logarithmic relationship to pitch, so it is used to convert frequencies into a Mel-frequency scale, which better represents how humans perceive pitch.

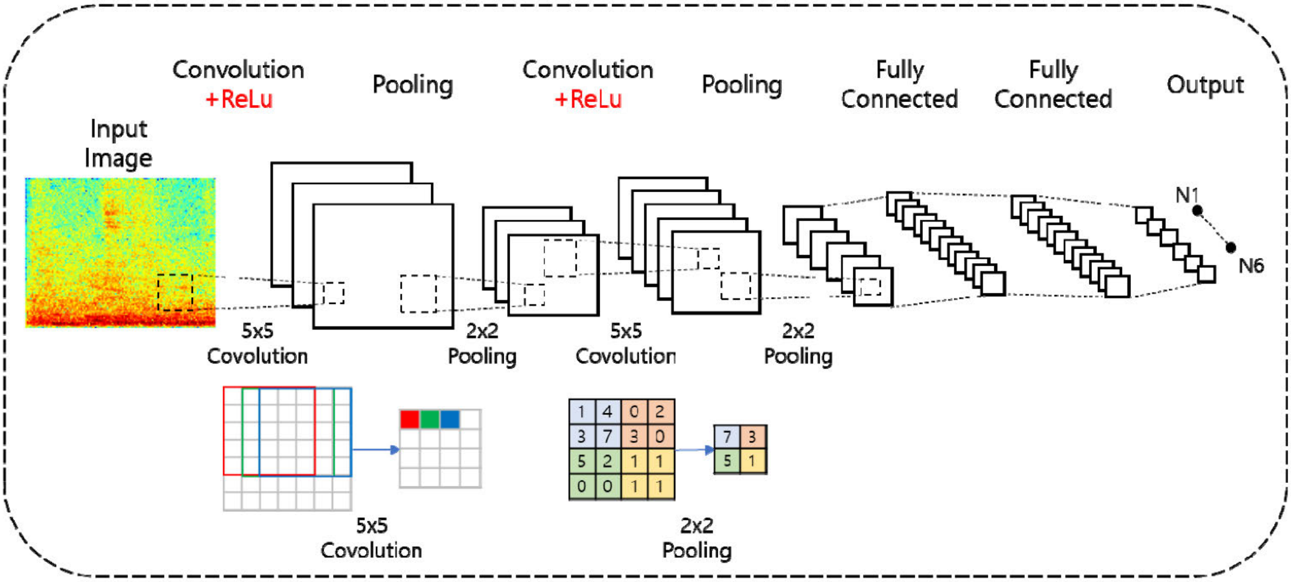


FIGURE 3. Structure of CNN [62].

In contrast to the STFT, the Continuous Wavelet Transform (CWT) does not rely on the dimensions of the analysis window and time-shift to establish time and frequency resolutions. Instead, the CWT employs a fundamental waveform known as a ''wavelet'' to facilitate the dissection of the speech signal. This method involves convolving the signal with shift and compressed iterations of the wavelet, achieved through temporal shifting.

$$CWT(u, s) = \frac{1}{\sqrt{S}} \int_{-\infty}^{\infty} X(t)\psi^{*}(\frac{t-u}{s})dt \qquad (10)$$

where, x(t) denotes the speech signal, u and s correspond to the shift and scale parameters, respectively, ψ represents the mother wavelet or base function, and * represent complex conjugate operation. In the context of study [57], the selected mother wavelet is the Morlet wavelet.

To summarize, raw 1D audio signals or spectrograms can be utilized as input to a CNN model.

CNNs are generally comprised of multiple convolutional layers, a Rectified Linear Unit (ReLU), pooling layers, fully connected layers (i.e., dense layers), and a Softmax layer as shown in Figure 3. A convolution layer is a layer in a DNN that applies a filter of convolution operation to the input signal to produce feature maps, and then passes the obtained feature maps to the next layer.

To extract the significant features from the obtained feature maps in a convolution layer, the ReLU activation function can be applied. In a ReLU, negative input values become 0 and the same input value remains for the non-negative numbers. Therefore, a ReLU enhances the non-linearity of the model, as well as, prevents over-fitting to the trained input samples. A ReLU is the most commonly used activation function in CNNs, although there are other activation functions such as sigmoid activation and, tanh activation and others.

A pooling layer is a layer in a CNN to reduce the size of the input while still retaining important information. This can be achieved by the pooling layers reducing the dimensions of the input feature map. This enables the model to focus on more important features, thus improve the accuracy. The most commonly used pooling layers are max and average pooling.

A fully connected layer in a CNN is used to connect all neurons in a previous layer to all neurons in the next layer. Fully connected layers generally come after the convolutional layer. A fully connected layer is a neural network layer with a fully connected configuration, and it is used to process the output of the convolutional layer after the flattening operation.

A softmax layer in a CNN is a type of output convolutional layer that uses a softmax activation function, which is a type of logistic sigmoid function used to turn arbitrary real-valued scores into a probability distribution. In the softmax layer, the output layer produces a probability distribution over the possible classes for a given input. This enables the model to predict which class a particular input signal belongs to on the basis of the probabilities it produces. It is also a good choice for multi-class classification problems because it produces probabilities that sum to 1, making it ideal for assigning a class label to an input.

A CNN is very useful to analyze and classify images, and has produces promising results for a variety of speech analysis applications since 1D audio signals are generally converted into 2D patterns (spectrograms) for the analysis using a CNN.

2) DEEP-LEARNING-BASED METHODS THAT USE CNNs

Shin et al. [59] used different CNN architectures for inter-floor noise classification. Features were first extracted from the audio signals using log-Mel spectrograms. The generated spectrograms were then input to various deep learning models such as ResNet, DenseNet, EfficientNet, ResNet, and Inception models. The performance of these models for noise classification ranged between 91.43–95.27%. These results indicated that the best accuracy (95.27±2.30%) was achieved with ResNet.

Another deep learning based method for inter-floor noise classification and position classification was proposed by Choi et al. [60]. They also investigated log-Mel spectrograms and different deep learning models (i.e. VGG16, AlexNet, ResNet50 V1). Experiments on Floor Management Center dataset indicated that VGG16 achieved the best classification accuracy of 99.5% and the best position classification accuracy of 95.3%.

Khamparia et al. [61] used spectrogram images with a CNN and a CNN with a tensor deep stacking network (TDSN) for environmental sound classification. A TDSN applies matrix multiplication of hidden output layers. Authors conducted experiments on two environmental sound datasets, ESC-10 and ESC-50. Results showed that the CNN achieved an accuracy of 77%, whereas the TDSN achieved an accuracy of 56% on the ESC-10 dataset.

Park and Lee [62] used a CNN and noise spectrogram images for environmental noise classification, which is input to hearing aids. They first collected ten types of noise sounds from living environments then converted noise audio signals into spectrogram images. They also applied a sharpening mask that enhances the sharp features as well as a median filter to the generated noise spectrogram images. Experiments on a self-recorded dataset showed that the best classification accuracy of 99.25% was obtained when both sharpening mask and median filter was applied.

Mu et al. [63] used a temporal-frequency attention-based CNN (TFCNN) for environmental sound classification. They first converted environmental sounds to images using log-Mel spectrogram. They also conducted an experiment on the frequency-band characteristics of different environmental sounds. They observed that specific frequency bands are inactive. To extract the active bands from log-Mel spectrograms, they applied a masking. Then input, the clipped spectrogram images to the TFCNN. Subsequently frequency-attention and temporal-attention models are applied to the spectrograms and combined their outputs before giving them to the CNN model. Using the combined time-frequency attention with the CNN, their model improved in performances on two public datasets, UrbanSound8k and ESC-50.

Al-Hattab et al. [64] used the Mel-Frequency Cepstral Coefficients (MFCC) of audio signals as input to a simple CNN for environmental sound classification. The proposed CNN model has only three layers. Through fine-tuning of input parameters, however, the lightweight CNN has an accuracy of 95.59%, and produces competitive results compared to deep-learning models on UrbanSound8k.

Salamon and Bello [65] proposed a CNN with data augmentation for environmental sound classification. They first converted audio signals to Mel-spectrogram images then attempted to alleviate the data scarcity problem by applying different data augmentation techniques such as time stretching, pitch shifting, and dynamic range compression. When data augmentation was combined with the CNN, they achieved the best results on UrbanSound8K with 79% classification accuracy.

Another study on environmental sound classification was introduced by Mushtaq et al. [66]. They utilized Mel-spectrogram images as input to different CNN models; a CNN model with 7 layers, a CNN model with 9 layers, ResNet-152, and DenseNet-161. They also applied data augmentation to increase sample features. Experiments conducted on three datasets, UrbanSound8k, ESC-10, and ESC-50. ResNet-152 achieved the best classification accuracy of 99.04% on ESC-10 and 99.49% on UrbanSound8k. DenseNet-161 obtained an accuracy of 97.57% on ESC-50.

Hershey et al. [67] used different CNN architectures for soundtrack classification. They investigated AlexNet, ResNet, VGG, and Inception models using log-Mel spectrograms of soundtracks. They conducted experiments on a video dataset consisting of 100M videos from Youtube with 30,871 video labels. The results indicated the effectiveness of various CNNs for soundtrack classification.

Cheng et al. [68] applied noise-spectral characteristics to analyze and classify modified loud exhaust sounds from vehicles passing by. They used STFT spectrograms to convert sound signals to images then AlexNet for classification. Experiments on a self-recorded dataset showed that their approach can classify cars passing by if (a) they have or (b) do not have modified loud exhaust with an accuracy of 96%.

Dong [69] proposed a CNN-based method for music genre classification. The author first divided music signals into segments. Then obtained Mel-spectrograms of the segments were input the CNN model for classification. Finally, the author combined the predictions of all segments to perform the classification task. On the GTZN music dataset, the method achieved a human-level accuracy of 70%.

Costa et al. [70] used a CNN for music classification on three music datasets, ISMIR 2004, LMD, and ethnic African music, with distinct characteristics. To assess suitability for different classifiers for music classification, they compared the CNN with other classifiers by fusing various hand-crafted features. Experiments demonstrated that the CNN performed significantly better than the other classifiers in different scenarios and achieved an accuracy of 92%.

Yang and Zhang [71] used a duplicated CNN for music genre classification. They used a Mel-scale spectrogram as input to the CNN. The duplicated convolutional layers extract different information of Mel-scale spectrogram images by applying various pooling layers (i.e. average, max). They then extract features and subsequently combined them for the final classification. They also added a residual connection between layers to improve classification accuracy. Experiments on GTZAN demonstrated that their approach achieve an improved accuracy of 90.7%.

Matocha and Zieliński [72] investigated CNNs with stereophonic signals as input for music genre classification. Authors utilized the spectrograms of two-channel stereo signals as input to different CNNs with various network architectures. They also compared the performance of different CNNs with traditional classifiers such as a SVM. Experiments on the FMA music dataset showed that two-channel stereo signals did not improve classification performance. These findings suggest that monaural signals can be more suitable to input to CNNs for music genre recognition. On the FMA dataset, the proposed method obtained an accuracy of 60%.

Abdoli et al. [73] presented a 1D CNN that classifies environmental sounds directly from raw audio signals. One of the challenges of using audio signals as input to a 1D CNN is varying the length of the audio signals. Authors addressed this problem by splitting a signal into overlapping frames by using a sliding-window technique. They tested various CNN architectures with different input sizes. The first layer of the CNN model was a convolutional layer with a gammatone filterbank to simulate the human auditory filter response. Experiments on UrbanSound8k showed that 1D CNN achieved an accuracy of 89%. Another advantage of their model is that it is more lightweight (reduced training times).

Mughal et al. [74] used a CNN with MFCC-spectrogram images for music genre classification of Urdu music from Pakistan. They investigated various CNN architectures; a CNN with batch normalization layers, a CNN with global average pooling, and VGG16. An Urdu music dataset was generated from YouTube videos. The results indicated that the CNN with batch normalization achieved the best classification accuracy of 92.6% compared with the other models.

Suo [75] used a CNN with STFT and Mel-frequency spectrogram images for music genre classification. They investigated VGG16 with different optimizers (Adam and stochastic gradient decent) on GTZAN with these spectrogram images. The results showed that the CNN achieved human level accuracy of 68%.

Ozer et al. [76] introduced a CNN-based method for sound classification under noisy conditions. They utilized spectrogram image features (SIFs) that can provide better performance in an ambient noise environment. They obtained SIFs by applying a series of processes: First, FFT spectrogram of the sound signals were obtained. Normalization and conversion to gray-scale was then applied to find linear quantized images. Subsequently, image re-sizing was applied and the obtained features were input to a CNN for classification. Experiments on RWCP Sound Scene datasets demonstrated that their method achieved an accuracy of 98.63%.

Lesnichaia et al. [77] proposed a CNN-based method for classifying different foreign-language English accents, particularly Germanic, Slavic, and Romance languages. They proposed using linear scale amplitude Mel-scale spectrograms that are more powerful for representing accent features compared with logarithmic Mel-scale spectrograms. They found that Mel-spectrograms with 64 bands are better suited for classification. The features extracted using the amplitude Mel-spectrograms were then input to a CNN for classification. Experiments on the Speech Accent Archive revealed that they achieved an accuracy ranging between 96.4 to 98.7% on different accents.

Malla [78] used a CNN for Kashmiri accent classification. The author first converted audio signals into Mel and MFCC spectrograms. Then input to the CNN for classification. To test the CNN, the author generated a dataset. When the CNN was combined with Mel spectrograms, the model achieved an accuracy of 98.66%.

Grahm [79] conducted experiments using spectrograms and CNNs to learn whether a sample English speech was spoken by a native English, Dutch, Japanese, Polish or French person. The experiments were conducted on the IViE corpus, a Cambridge English Corpus using the LeNet CNN model. The results indicated that CNNs can identify the background language with high accuracy.

Hussain and Haque [80] introduced a lightweight and fast 1D-CNN called SwissNet for speech/music/noise classification. SwissNet uses MFCC spectrograms as input to the 1D-CNN and contains two parallel blocks of convolutional layers with gated activation (sigmoid and tanh functions); one block uses traditional convolutional layers and the other block use separable convolutional layers that can be trained faster. The features extracted from both parallel blocks are then combined and extra convolutional layers are applied. Evaluations on the MUSAN corpus and GTZAN demonstrated that SwissNet achieved high classification accuracy (above 97%).

Salehghaffari [81] proposed a CNN-based method for speaker/non-speaker verification. This method trains a CNN in an end-to-end manner to identify background features from the speaker's speech. The author used the Siamese model and tested the method on the VoxCeleb dataset.

Zaman et al. [82] used a CNN with STFT spectrogram images for harmful speech classification in hearing aid devices. The STFT spectrogram of the speech was first obtained then classified into clean speech and five different noise types using CNN models with varying complexity. Experiments on a custom dataset generated from the Commonvoice Mozilla dataset showed that their model can correctly classify harmful speech with an accuracy of up to 99%.

Ballesteros et al. [54] proposed a CNN called Deep4SNet, for fake-voice recognition. They separately trained fake and original speech histograms using Deep4SNet. They also trained the same speech data by using a machine learning model using hand-crafted features. On a custom dataset, their model achieved an accuracy of 98.5%.

Vrebcevic et al. [83] proposed a CNN with a spectrogram for emotion classification. They first converted speech signals into spectrogram images and applied various data augmentation techniques, such as down sampling and noise, to increase the training data size. They evaluated their model on the Berlin database.

Si et al. [84] proposed using a CNN to handle audio classification on low-resourced datasets such as those with low training data prone to over-fitting. They investigated the variational information bottleneck (VIB) to suppress irrelevant features using the CNN. They conducted experiments on various audio datasets and yielded substantial improvements in classification accuracy of up to 5.0% in low-source settings compared with baseline models. VIB is adaptable and can be easily used with other advanced network architectures.

Hussain et al. [85] compared a DNN and CNN for acoustic scene classification. The contribution of their work lies in combining the features of MFCC and log-Mel spectrograms as input to the DNN and CNN. Experiments on the DCASE (Detection and Classification of Acoustic Scenes and Events) 2017 dataset demonstrated that the combination of MFCC and log-Mel spectrogram features outperformed the compared models; The DNN achieved an accuracy of 83.45% and the CNN achieved an accuracy of 83.65%.

Dang et al. [86] developed a CNN with multi-scale spectrogram images for acoustic scene classification. They converted speech signals into MFCC and log-Mel spectrograms. They then concatenated these two spectrograms and fed to them to the CNN for classification. Experiments on the TUT Acoustic Scenes 2016 dataset showed that these multi-scale features improved the accuracy to 85.9% comparing with baseline methods.

Nguyen and Pernkopf [87] proposed an ensemble CNN for acoustic scene classification using the nearest neighbor filter. They first converted speech signals into different log-Mel spectrograms using the nearest neighbor filter or audio chunks without overlap. They then input these spectrograms to a series of different CNNs for feature extraction. Finally, they ensemble the extracted features for classification. Experiments on the DCASE 2018 Challenge demonstrated that their model with the nearest neighbor filter for feature extraction was significantly good.

Vafeiadis et al. [88] investigated two models for acoustic scene classification: A hybrid SVM-HMM and a CNN. In the CNN, log-Mel spectrograms were normalized and then fed to the model. Data augmentation techniques were also used to increase the size of the training samples. The CNN provided promising results compared with the baselines.

Pham et al. [89] proposed an ensemble CNN for acoustic scene classification. They combined three spectrograms into one image that was used for the training of their ensemble CNN. Specifically, log-Mel, gammatone filter and constant Q spectrograms were combined using an add layer, convolutional layer, and DNN blocks. In the ensemble CNN, feature extraction was executed using the CNN, then the extracted features were input to two DNN blocks for classification. Experiments on DCASE 2016 demonstrated that their model was effective (90%), significantly outperforming traditional models.

Jena et al. [90] proposed a multi-modal CNN architecture for music genre classification. They specifically used two types of speech inputs; spectrogram and wavelets. Inside the CNN model, features extracted from multi-modal inputs and fused for classification. Experiments on GTZAN and Ballroom datasets demonstrated that their architecture with multi-modal inputs achieved an accuracy of 81% on GTAZ and 71% on Ballroom datasets.

Arias-Vergara et al. [57] proposed a new time-frequency representations of audio signals by combining continuous wavelet transform, Mel-spectrograms, and Gammatone spectrograms to form a new 3D-channel spectrograms for the applications to analyze speech in automatic detection of disorder speech of cochlear implant (CI) users. During the training of CNN model speech signals from both cochlear implant (CI) users and HC were used while using the proposed new time-frequency representations to conduct binary classification. Based on the findings, the best performance was achieved when training CNN with 3D-channel spectrograms extracted from off-set transitions.

Lopac et al. [175] also utilized different time-frequency representations and proposed a method that uses Cohen's time-frequency representations (TFRs) and deep learning algorithms (CNNs) to classify noisy non-stationary time-series signals. Subsequently, computed 12 distinct time-frequency representations (TFRs) from Cohen's class using the original noisy time-series data. These TFRs served as the input to train three convolutional neural network architectures such as ResNet-101, Xception, and EfficientNet. The classification results obtained from this approach significantly improved as compared to those achieved by a baseline model trained.

B. RNN-BASED METHODS

In this section, we discuss RNN-based methods for audio classification.

1) RNN ARCHITECTURES

RNNs are a type of neural network (NN) that can learn from the temporal context of a sequence of data. Traditional NNs have the problem of gradient vanishing. For sequences of data, NNs forget the past input. RNNs tackle this problem by using recurrent processing of data, where the output of the NN is also input to the NN, and these NNs are called Recurrent NNs (RNNs). Therefore, RNNs can take sequence of data as input and process one element at a time and extract features. This enables them to capture the temporal context of data, such as a sentence or a time series. RNNs are used in a variety of tasks including image captioning, language translation, language modeling, and speech recognition.

RNNs can be used to classify audio signals, such as detecting speech or music in a sound recording. In this case, RNNs are trained to recognize patterns in the audio signal over time. By processing audio frames sequentially, RNN can learn to recognize the characteristics of different sounds, enabling it to accurately classify the audio. RNNs enable computers

TABLE 1. Comparison of CNN-based methods.

Reference /Years	Method	Input Data/feature sets	Deep Learning Architecture	Dataset	Performance	Application
[59] 2020	CNN	Log-Mel spectrogram	DenseNet, ResNet, Inception, EfficientNet	Self-recorded	95.27±2.30%	Inter-floor noise classification
[60] 2019	CNN	Log-scaled Mel-spectrogram	VGG16, AlexNet, ResNet	SNU-B36-50	99.5%	Inter-floor noise classification
[61] 2019	CNN	Spectrogram	CNN, TDSN	ESC-10 and ESC-50.	77%	Environmental sound classification
[62] 2019	CNN	Spectrogram (sharpening mask and median filter)	CNN	Self-record	99.25%	Environmental noise classification
[63] 2021	CNN	Log-Mel spectrogram	TFCNN	ESC-50 UrbanSound8K	84.4% and 93.1%	Environmental sound classification
[64] 2021	CNN	MFCC	CNN	UrbanSound8k	95.59%	Environmental sound classification
[65] 2017	CNN	Mel-spectrogram plus data augmentation	CNN	UrbanSound8k	79 %	Environmental sound classification
[66] 2021	CNN	Mel spectrogram	CNN, ResNet, DenseNet	ESC-10, ESC-50 and UrbanSound8k datasets.	99.04% 97.57%	Environmental sound classification
[67] 2017	CNN	Log-Mel spectrogram	AlexNet, VGG, Inception, and ResNet	YouTube-100M		Soundtrack classification
[68] 2023	CNN	Spectrograms	AlexNet	Self-recorded	96%	Modified loud exhaust classification
[69] 2018	CNN	Mel-spectrogram	CNN	GTZN datasets	70%	Music genre classification
[70] 2017	CNN	Spectrogram	CNN	ISMIR 2004, LMD, ethnic African music	92%	Music classification
[71] 2019	CNN	Mel-Scaled spectrogram	Duplicated CNN	GTZAN dataset	92%	Music genre classification
[72] 2018	CNN	Stereophonic signals spectrogram	CNN	FMA datasets	60%	Music genre classification
[73] 2019	CNN	1D audio signal	CNN	UrbanSound8k	89%	Environmental sound classification
[74] 2022	CNN	Mel-Frequency Cepstral Coefficients	CNN, VGG16	Custom dataset from YouTube	92.6% .	Music genre classification
[75] 2022	CNN	STFT and the Mel-frequency spectrum	CNN	GTZAN dataset	68.3%	Music genre classification
[76] 2018	CNN	Spectrogram image features	CNN	RWCP Sound Scene Database	98.63%	Sound classification
[77] 2022	CNN	Mel-scale amplitude spectrograms	CNN	Speech Accent Archive	Ranging 96.4% to 98.7%	English accent classification
[78] 2022	CNN	MFCC and Mel-spectrograms	CNN	Custom	98.66%	Kashmiri accent classification
[79] 2021	CNN	Spectrogram	LeNet	IViE corpus, Cambridge English Corpus	Above 90%	Background language classification
[80] 2018	CNN 1D	MFCC spectrograms	CNN (SwissNet)	MUSAN corpus and GTZAN	Above 97%	Speech/Music/ Noise classification
[81] 2018	CNN	MFCC spectrograms	CNN	VoxCeleb dataset	ERR 10.5%	Speaker/ Non speaker verification
[82] 2020	CNN	STFT spectrograms	CNN	Commonvoice mozilla	99%	Harmful speech classification
[54] 2021	CNN	Histograms	CNN	Custom	98.5%	Fake voice recognition
[83] 2019	CNN	Spectrograms	CNN	EmoDB corpus	58%	Emotion classification
[84] 2021	CNN+ VIB	MFCC	CNN	MNIST, ESC-50, Toronto Emotional Speech Set (TESS),TUT	99.8.%	Audio classification
[85] 2018	CNN	MFCC and Log-Mel energies	CNN	DCASE 2017	83.65%	Acoustic scene classification

TABLE 1. *(Continued.)* Comparison of CNN-based methods.

[86] 2018	CNN	Combined Log-Mel and MFCC spectrograms	CNN	TUT Acoustic Scenes 2016	85.9%	Acoustic scene classification
[87] 2018	CNN	Log-Mel spectrogram	Ensemble CNN	DCASE 2018	69.3%	Acoustic scene classification
[88] 2017	CNN	Log-Mel spectrogram	CNN	DCASE 2017	95.9%	Acoustic scene classification
[89] 2019	CNN	Log-Mel filter, Gammatone filter, and Constant Q transform	Ensemble CNN	DCASE2016	90%	Acoustic scene classification
[90] 2023	CNN	Spectrogram and Wavelet	CNN (Multimodal input)	GTZAN, Ballroom	81%	Music genre classification
[57] 2020	CNN	3D channel spectrogram	LeNet-5 Convolutional Network	CI Speech Self-recorded	84%	Automatic detection of disordered speech of CI users
[175] 2022	CNN	Cohen's class time-frequency representations (TFRs)	ResNet, Xception, and EfficientNet	LIGO data	97.10%	Noisy non-stationary signals classification

to understand the content of audio data due to their ability to capture the temporal context of an audio signal. RNNs can be used to classify audio signals by extracting features from the signal, such as the frequency, amplitude, and phase. Additionally, RNNs can be used to classify music genres by analyzing the acoustic features of a song, such as rhythm, pitch, and timbre.

Although RNNs are successful for data sequences, such as speech signals, they have a short-term memory. In different cases, we might need short-term dependencies or long-term dependencies. For longer sequences of inputs, Long Short-Term Memory (LSTM), which is a type of RNN, is used to remember longer-term dependencies in the input data. In LSTM, instead of a single NN, there are four layers, interacting with each other to keep relevant information in the cell state (memory), as shown in Figure 4. These four layers are: Cell state, forget gate, input gate, and output gate. The cell state transfers the vector coming from previous hidden layers to the next LSTM layer. The forget gate is a NN with sigmoid activation to determine which current input to include or forget (remove) from the cell state by using vector multiplication. The input gate contains two NNs; one with the sigmoid activation and the other with the tanh activation. NN with the tanh activation selects the candidate inputs to be included after applying matrix multiplication with the output of the NN with the sigmoid activation. The output vector is then added to the cell state by applying vector addition. The output gate is responsible for producing an output vector with the NN with sigmoid activation. With the recurrent configuration of LSTM layers (output of one LSTM layer is an input to the other), long-term dependencies can be learned from sequences of input data. LSTM has been applied to many speech classification tasks.

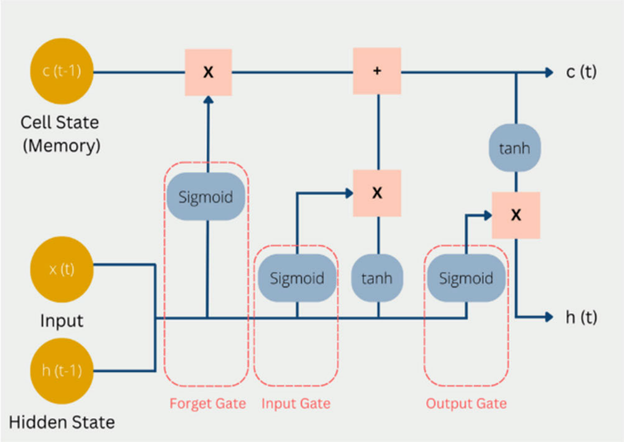


FIGURE 4. LSTM cell architecture.

2) DEEP-LEARNING-BASED METHODS THAT USE RNNs

Scarpiniti et al. [91] proposed a deep RNN (DRNN) using recurrent LSTM for construction site audio recording classification. The input of the DRNN is composed of multiple spectral features such as MFCCs, Mel-scaled spectrograms, chroma, and spectral contrast. It achieved an overall accuracy of up to 97% on the test set and surpassing other models.

Yu et al. [100] presented a Bi-RNN model with an attention mechanism. They also implemented two different attention-based models, serial and parallelized, to compare their performance using STFT spectrograms. The results indicated that the parallelized attention model is more effective and yields better results than the serial attention model.

Gan [101] introduced an RNN model with a channel attention module for classifying music features. This model uses a combination of GRUs and Bi-RNNs, such as Bi-LSTM, and utilizes an attention mechanism to allocate varying attention weights to the outputs of the RNN at diverse points. This is to more accurately capture the general characteristics of the music. Proposed RNN achieved a classification accuracy of 93.1% on GTZAN and AUC score of 92.3% on the

TABLE 2. Comparison of RNN-based methods.

Reference /Years	Method	Input Data/feature sets	Deep Learning Architecture	Dataset	Performance	Application
[91] 2020	RNN	MFCCs, Mel-scaled Spectrogram, Chroma, and Spectral contrast	LSTM	Self-recorded	97%	Audio classification in construction sites
[100] 2020	Bi-RNN	STFT	Bi-RNN	*GTZAN and Extended Ballroom*	**92.7%**	Music genre classification
[101] 2021	Bi-RNN	Sound spectrum	Bi-RNN (Bi-LSTM)	GTZAN and MagnaTagATune	92.3%	Music feature classification
[111] 2021	RNN	MFFC for audio fingerprint creation	LSTM	UrbanSound8K	98.8%	Environmental sound classification

MagnaTagATune multi-label labeling dataset, which is better than other compared models.

Banuroopa and Shanmuga Priyaa [111] proposed a fingerprinting approach for audio signals by using the mean of the MFCC spectrum. The spectrum is then converted to a binary image and input to a LSTM network for classifying environmental sound. Experiments on UrbanSound8K revealed that the processed and modified fingerprint images with LSTM achieved an accuracy of 98.8%.

C. AUTOENCODER-BASED METHODS

In this section, we discuss autoencoder-based methods for audio classification.

1) AUTOENCODER ARCHITECTURES

An autoencoder is an unsupervised learning technique that uses ANNs to learn compressed data representations (encodings) of unlabeled data. It consists of two steps (Figure 6): An encoder first transforms the input data into a lower-dimensional representation, then a decoder recreates the output data from the encoded input representation. Autoencoders use backpropagation to learn an encoding of input data that can be used to recreate input representation with minimal information loss. This encoding is referred to as the "latent representation" or "latent space" of the data. Autoencoders are effective in learning important features of a dataset with a reduced feature set. In addition, autoencoders are also effective in detecting outliers and anomalies in the data.

Autoencoders are types of ANNs that can be used to learn features from audio data, such as frequency components or temporal patterns. Autoencoders first encodes the audio features into a compressed representation, then uses the encoded features to train a classification model. The model can then be used to classify audio signals on the basis of their compressed representations. Autoencoders are especially useful for audio classification tasks with high-dimensional input data, such as speech classification or music classification. They can also be used to identify subtle changes in audio samples, such as changes in pitch or tempo, which can then be used to distinguish between different classes of audio samples. As a pre-processing step, autoencoders can be utilized before other deep-learning models, such as CNNs, to improve classification accuracy. Autoencoders can also be used to decrease input data complexity, which can help reducing the amount of time and resources needed for training and inference.

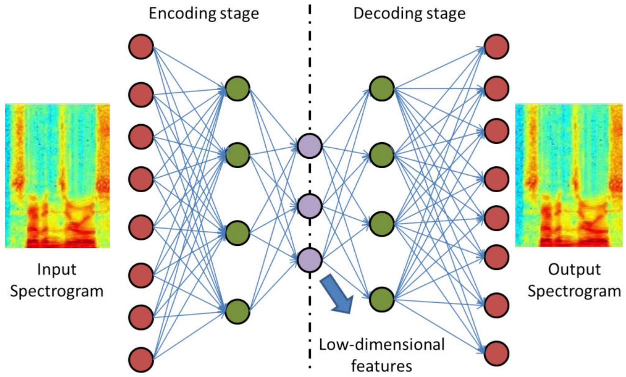


FIGURE 5. Autoencoder architecture [113].

To conclude, autoencoders are NN architectures used for unsupervised learning and data compression. They take an input dataset and learn the essential features to create a compressed representation by using fully connected, convolutional, and pooling layers, which reduces the dimensionality and complexity of the input. This representation is then reconstructed using decoding layers and output to an output layer to generate a reconstructed output that closely matches the original input.

2) DEEP-LEARNING-BASED METHODS THAT USE AUTOENCODERS

Amiriparian et al. [115] proposed a recurrent autoencoder architecture for acoustic scene classification. They first generated different Mel spectrograms from audio signals then input separate spectrogram sets into parallel autoencoder networks. Finally, they fused the extracted features from different autoencoders then feed them to the MLP for classification. They conducted experiments on DCASE 2017 and showed that their architecture is effective.

Huang et al. [116] applied image masked autoencoders (known as MAEs) to audio signals for speech classification.

TABLE 3. Comparison of autoencoder-based methods.

Reference /Years	Method	Input Data/feature sets	Deep Learning Architecture	Dataset	Performance	Application
[115] 2017	Autoencoder	Mel-spectrograms	Parallel autoencoders with MLP	DCASE 2017 Challenge	88%	Acoustic scene classification
[116] 2022	Autoencoder	Mel-spectrograms	Audio MAE	AudioSet,ESC-50,SPC-SPC-1, SID)	98.3%	Speech classification
[118] 2021	Autoencoder	MFCC	DSP methods with Autoencoder	GTZAN dataset	81.9%	Music genre classification
[119] 2021	Autoencoder	MFCCs and latent variables	FHVAE	Torgo and COPAS	--	Speech disorder classification
[121] 2021	Autoencoder	Raw data	Autoencoder with feedforward classifier	GTZAN	94%	Music genre classification
[123] 2022	Autoencoder	MFCC	Different variational autoencoders	ICBHI sound datasets	<90%	Respiratory disease classification

Authors used Mel spectrograms to represent raw audio signals then applied different masking strategies to divide Mel spectrograms into different patches, such as time, frequency, and time-frequency patches. Both encoder and decoder networks consist of transformers, particularly standard 12-layers vision transformers. They conducted experiments on four datasets. Key findings of the experiments were that MAE performed surprisingly well for audio spectrograms, stronger representations can be learned with local self-attention in the decoder, masking can be executed during pre-training, and fine-tuning can help increase accuracy and decreasing training time.

Atahan et al. [118] introduced a combined autoencoder and digital signal processing (DSP) method for music genre classification and recommendation. They applied various DSP methods for feature extraction on GTZAN then generated and input MFCC spectrograms in an autoencoder for feature extraction for music genre classification. Finally, they applied different classifiers, such as an SVM, Random Forest (RF), and MLP, for the classification. The results from this process were then used for music recommendation.

Qi and Van hamme [119] used factorized hierarchical variational autoencoders (FHVAEs) for speech disorder classification. They extracted both sequence-based latent variables and content-based data for disorder representation. Evaluations on Torgo and COPAS datasets showed that better results are achieved when latent variables are combined at sentence and word level.

Sawhney et al. [121] considered the idea of learning a latent representation of musical genre from raw input audio by using hybrid neural networks with both autoencoding and classification components. After the training, MLP is used for classification. Authors found that musical genre is closely related to style, suggesting that such a representation can be learned directly from audio.

Saldanha et al. [123] aimed to provide a solution to the imbalanced dataset problem for respiratory disease classification. They investigated the use of different variational autoencoders (VAEs) for synthesizing respiratory sounds for imbalanced classes. They compared the Multilayer Perceptron VAE, Convolutional VAE, and conditional VAE for data augmentation to improve classification accuracy. They conducted their experiments on ICBHI sound datasets, and the results showed that VAEs can improve the accuracy when using data augmentation.

D. TRANSFORMERS AND SPECTROGRAM TRANSFORMERS

In this section, we discuss transformers for audio classification.

1) TRANSFORMER ARCHITECTURES

Transformers are a type DNNs with an attention mechanism. Transformers were initially designed for NLP for language models [131] then applied to images using vision transformers. Recently, transformers have been applied to audio signals using spectrogram transformers. In particular, new transformer-based methods have been introduced using an attention mechanism to significantly improve audio classification by enabling the model to be aware of the global context [135]. Transformer-based methods, compared with CNN-based methods, can handle input-length variance, which is one of the advantages of transformers. This can be achieved due to the ability of the multi-head self-attention mechanism to work with variable lengths of input sequences. Therefore, transformer-based methods can promptly capture useful global-context information, regardless of the audio length. Transformers are generally referred as data hungry since they require a large amount of training data. In cases where labeled data are limited, many audio transformers use pre-training models and fine-tuning. The Patchout faSt Spectrogram Transformer (PaSST) [133] and Audio Spectrogram Transformer (AST) [132] are two of the leading models for audio classification. The AST [132] is the first transformer model for audio classification and adapts pre-training weights from the image classification network of vision transformers (ViTs) [134]. The PaSST reduces the computation and memory complexity of training transformers for the audio domain.

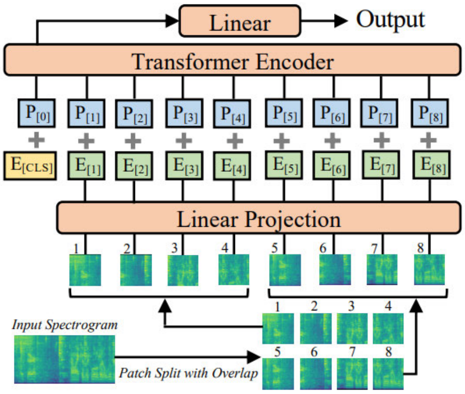


FIGURE 6. Architecture of transformer [137].

The use of transformers for audio classification has gained attention due to their promising results. To use transformers effectively, the following steps are generally taken into consideration. First audio signals are converted into visual spectrograms using feature extraction techniques such as using STFT, Mel-spectrograms, and log-Mel spectrograms and MFCCs. Spectrograms contain time-frequency information and serve as the input to a transformer. Next, spectrograms are often divided into small, fixed-length segments known as ‘‘patches’’. These patches are then treated as individual input tokens, similar to the words in NLP. Then, the transformer architecture is composed of a stack of encoder layers, each consisting of self-attention mechanisms and feed-forward neural networks, as shown in Figure 7. The original transformer models do not have convolutional layer but contain feed-forward layers. A residual connection and layer normalization are then applied. The self-attention mechanism helps the model capture the dependencies between different patches in the spectrogram, the same as with the original transformer model. Global average pooling and max pooling are common techniques used to aggregate information across the time dimension after the encoder layers and provide a fixed-length representation of the audio. This reduces the temporal dimensionality of the input audio. The pooled representation is then fed into one or more dense layers (fully connected layers). The output of these layers is then passed through a softmax activation function, which produces class probabilities as the output. Finally, by pre-training on large amounts of unlabeled audio data using techniques such as self-supervised learning or contrastive learning, transformers for audio classification can gain useful audio representations. These representations can then be used to fine-tune the model on specific audio classification tasks using labeled data.

2) DEEP LEARNING METHODS THAT USE TRANSFORMERS

Zhang et al. [135] presented a spectrogram transformer that is a combination of various feature extraction strategies for environmental sound classification. They first converted the audio signals into spectrogram images using FFT. Then applied various attention blocks to enhance the extracted features from the time and frequency domains using a transformer encoder. They tested different attention mechanisms and obtained the best results using the temporal-frequency attention block on ESC-50 without pre-training.

Luo et al. [136] investigated the impact of the patch-level feature fusion approach using ViTs for different audio classification tasks. They first obtained a Mel spectrogram of the audio signal and divided it into patches then input these patches to the ViT encoder for feature fusion. From the fused patches, new patches were then generated and input to an MLP for classification. They investigated the effectiveness of this approach on various classification tasks using ImageNet and AudioSet pre-trained model weights.

Nogueira et al. [53] conducted a comprehensive analysis of using transformers against different baseline CNN models for urban sound classification. They investigated the performance of a transformer, baseline CNN, DenseNet, Inception-V3, and pre-training options together with data augmentation. As a transformer, they used the spectrogram transformer proposed by [137] for comparison. Experiments on UrbanSound8K, ESC-50 and ESC-10 demonstrated that a transformer model using transfer learning from AudioSet achieved the best accuracy.

Elliott et al. [138] tackled the problem of efficiency versus large parameters. Many deep learning models, such as CNNs for audio classification, require large parameters, meaning large storage for testing. However, many microcontrollers do not have such memories to operate. To overcome this problem, they proposed a tiny transformer for audio classification using a BERT-based transformer (designed for language model) that is trained on Mel-spectrogram images. They also investigated various data augmentation techniques. The proposed tiny transformer contains around 6000 parameters and achieved an accuracy of 99.85% for environmental sound classification on ESC-50.

Gong et al. [139] improved upon their previous study on an AST with a self-supervised learning strategy. Although transformers are effective, they require large amounts of training samples to learn the feature maps. In small domains where the training samples are limited, transformers might not be as effective as CNNs. Therefore, authors tried to alleviate this problem by integrating a self-supervised learning framework into the AST. They proposed a generative model that learns from masked spectrogram patches. Experiments showed that the proposed model improved the accuracy on unlabeled audio data from the AudioSet and Librispeech datasets.

Koutini et al. [140] focused on the problem of efficient training of transformer models. Since transformers require large amounts of training audio samples, and the complexity of the training increases quadratically as the input increases. To alleviate this, they presented a novel approach for optimizing and regularizing transformers on audio spectrograms using a patchout spectrogram transformer to provide efficient training. Experiments on Audioset demonstrated that their patchout spectrogram transformer outperform CNNs in both performance and training speed.

Zhao et al. [141] proposed a swin transformer that contains a self-supervised pre-training method for music genre classification. Swin transformers merge smaller patches as they go deeper in the architecture. They first converted audio signals into spectrograms and applied data augmentation Then applied self-supervised pre-training using transfer learning and pre-training. They input these patches into the swin transformer. Experiments on GTZAN demonstrated that swin transformer improved the accuracy.

Due to the data hungry nature of transformers, many transformers for audio classification use pre-trained models from the image domain such as ImageNet. Atito et al. [142] tackled this problem and proposed a self-supervised transformer called ASiT, which reduces the dependency on pre-trained models from the image domain. Specifically, general audio representations are obtained with local and global context by applying group-masked model learning and self-distillation. They evaluated ASiT for audio/speech classification and achieved state-of-the-art performances.

Liu et al. [143] presented the Causal Audio Transformer (CAT), a model that utilizes Multi-Resolution Multi-Featured (MRMF) feature extraction with an acoustic attention block for improved audio modeling. The CAT also includes a causal module to reduce over fitting, facilitate knowledge transfer, and enhance interpretability. Experiments show that the CAT achieved superior or comparable results compared to the other models on ESC50, AudioSet, and UrbanSound8K and can be easily adapted to other transformer-based models.

Zhuang et al. [145] proposed one of the early transformer-based method for music genre classification. They first converted the audio data to log-amplitude Mel spectrograms and applied the original multi-head attention transformer to GTZAN. The results indicated that transformers can be used for music genre classification.

Qiu et al. [146] proposed a bi-directional transformer with a masked predictive model for music genre classification. The transformer also uses a pre-processing called Pitch-to-Vector (Pitch2Vec) that converts audio signals into vector sequences. The masked predictive encoder then extracts bi-directional representations about the music with an unsupervised learning strategy. Experiments demonstrated that the transformer can achieve good accuracy.

Verma and Berger [147] proposed a transformer-based architecture to process raw audio signals without the need for convolutional layers for audio classification on the FSD50K dataset. Compared with ASTs, in this approach raw waveforms are input to the transformer for the classification task. They investigated the performance of the transformer-based architecture by comparing it with that of a CNN. The advantage of their architecture is that no unsupervised pre-training is used as well as, pooling techniques from CNNs and multi-rate signal processing ideas from wavelets are used.

Chen et al. [148] proposed a hierarchical token semantic audio transformer (HTS-AT) to tackle the problem of high memory/long training time as well as the need for pre-trained models from the image domain. HTS-AT uses a hierarchical structure to reduce the amount of require memory and training time. It also uses a semantic module to map into class feature maps for audio event detection. Evaluations on AudioSet and ESC50 demonstrated that HTS-AT can achieve state-of-the-art performance on speech-command recognition.

Primus and Widmer [149] investigated the performance of the PaSST against two CNNs in zero-shot learning settings. In zero-shot learning, the model attempts to predict unseen classes with adaptable class representations. In particular, authors investigated the performance of PaSST with two CNN models (VGG and a custom CNN model). Experiments on three datasets, namely AudioSet, ESC-50 and OpenMIC, illustrated that the PaSST outperformed CNN counterparts in zero-shot learning in all tasks.

Ghosh et al. [150] integrated multi-scale feature hierarchies into an AST called a multi-scale AST (MAST) for audio classification. This MAST involves multiple patchifying (dividing the spectrograms into patches); as the network goes deeper, new patches are generated and the patch sizes increase. Thus, the number of patches decreases. Therefore, a pyramidal structure is obtained. The initial MAST layers handle high temporal resolution/low embeddings, whereas the deeper layers capture high-level information. Experiments demonstrated that MAST performed better than an AST.

Another MAST was proposed by Zhu and Omar [151] that incorporates a hierarchical learning model into an AST. This MAST uses both 1D and 2D pooling operations to reduce feature dimensions and the number of tokens. Experiments on various datasets demonstrated that proposed MAST is effective for different audio classification tasks.

Akbari et al. [152] proposed a convolution-free transformer-based framework that learns representations from multi-modal data, such as video, audio and text, in a kinetics environment. This framework receives data from video, audio, and text, and fuses them in multi-modal representations using a transformer from unlabeled data. Audio data are input as raw wavelets and audio data are fine-tuned in accordance with AudioSet. They conducted several experiments to test the efficiency of their framework, revealing that it is effective.

E. HYBRID MODELS

In this section, we discuss hybrid models. According to the reviewed methods, we observed that hybrid models can be divided into two categories: (1) Hybrid deep learning models that combine various deep learning architectures such as CNN-LSTM, CNN-transformers and others. (2) Hybrid deep

TABLE 4. Comparison of transformer-based methods.

Reference /Years	Method	Input Data/feature sets	Deep Learning Architecture	Dataset	Performance	Application
[135] 2022	Transformer	Spectrogram (FFT)	Audio Spectrogram Transformers	ESC-50 dataset	57.24%	Environmental Sound classification
[136] 2022	Vision Transformer	Mel-spectrogram	Vision Transformer with patch level feature fusion	ESC-50 dataset, SCV2 dataset, CREMA-D dataset	98.13%	Various classification tasks: Environmental sound classification, speech command recognition and etc.
[53] 2022	Transformer	Spectrogram	Audio Spectrogram Transformers	UrbanSound8K, ESC-50, ESC-10 datasets	99%	Urban sound classification
[137] 2021	Transformer	Spectrogram	Audio Spectrogram Transformers	ESC-50, Speech Command V2	98.1%	Environmental Sound Classification and speech command Classification
[138] 2021	Transformer	Mel-spectrogram	Tiny transformer (BERT-based)	ESC-50	99.85%	Environmental sound classification
[139] 2022	Transformer	Spectrogram	Audio Spectrogram Transformers with self-supervised learning	AudioSet and Librispeech	Average improvement of 60.9% on different tasks	Various audio/speech classification tasks
[140] 2021	Transformer	Spectrogram	Patchout Fast Spectrogram Transformer	Audioset	---	Various audio classification tasks
[141] 2022	Transformer	Spectrogram	Swin transformers	GTZAN	97.2%	Music genre classification
[142] 2022	Transformer	Spectrogram	Audio Spectrogram Transformers with self-supervised learning	Speech Commands	98.8%	Audio event classification, keyword spotting, and speaker identification
[143] 2023	Transformer	Spectrograms	Causal Audio Transformer	AudioSet, ESC50, UrbanSound8K	97.2%	Various audio/speech classification tasks
[145] 2020	Transformer	Log-amplitude Mel-spectrogram	Transformer	GTZAN dataset	76.0%	Music genre classification
[146] 2021	Transformer	Pitch to vectors	Bidirectional transformers using masked predictive encoder	Lakh MIDI dataset	94%.	Music genre classification
[147] 2021	Transformer	Raw waveforms	Transformer	FSD50K	53.7%	Audio classification
[148] 2022	Transformer	Mel-spectrogram	Hierarchical Token Semantic Audio Transformer	AudioSet ESC-50 Speech Command V2	97% and 98%	Audio classification and detection
[149] 2022	Transformer	Mel-spectrogram	Patchout Fast Spectrogram Transformer	AudioSet, ESC-50, OpenMIC	80.47%	Various audio/speech classification tasks
[150] 2022	Transformer	Spectrogram	Multiscale Audio Spectrogram Transformers	VoxCeleb (VC), Speech Commands (SC) v1 v2, VoxForge (VF) , IEMOCAP (IC) IEMOCAP (IC), NSynth US8K	97.4%	Audio classification
[151] 2023	Transformer	Spectrogram	Multiscale Audio Spectrogram Transformers	Kinetics-Sounds, Epic-Kitchens-100, VGGSound ,AudioSet	81.3%	Various audio/speech classification tasks
[152] 2021	Transformer	Raw waveforms	Transformers with Multimodal Data	AudioSet	97.1% and mAP of 39.4%	Various classification tasks

learning models with a traditional classifier, where feature extraction is performed by a deep learning model and the classification is achieved by a traditional classifier like SVM, KNN and others. According to this categorization, methods are reviewed under these hybrid deep learning categories as follows.

1) HYBRID DEEP LEARNING MODELS

In the literature, we found cases of hybrid models that various deep learning architectures can be combined to create more powerful audio classification models. The aim of these hybrid models is to combine the strengths of different deep learning models. For example, many RNN methods initially use CNN for extracting features from spatial inputs such as spectrogram images, and then output of the CNN is given to the RNN network like LSTM. While the recurrent layers are trained to identify patterns in temporal inputs such as audio or time series data. A sample CNN-LSTM architecture is given in Figure 7. Similarly, autoencoders can be combined with RNNs (i.e. LSTM) or CNNs. Transformers can also be combined with CNNs. These combination of models enables efficient processing of both spatial and temporal information. These hybrid models are used in different applications such as music genre classification, environmental sound classification, acoustic scene classification and so forth.

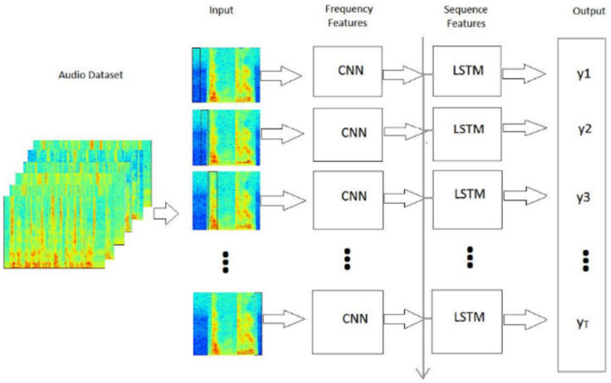


FIGURE 7. Sample CNN-LSTM architecture [112].

a: DEEP LEARNING METHODS THAT USE HYBRID MODELS

Sang et al. [92] presented a CNN-LSTM network for urban-sound classification that takes time-domain waveforms as input. This model integrates CNN and a two layer LSTM model to extract sound features and temporally aggregate them. Experiments on UrbanSound8k dataset indicated that their model performed well.

Liao et al. [93] proposed an effective sequential CNN-LSTM architecture for automatic music classification. The output of the CNN is fed to a two-layer sequential LSTM layers. Proposed architecture demonstrated a higher classification accuracy of 92.1%, outperforming the baseline models of CNN and RNN alone.

Asatani et al. [94] developed an approach to categorize respiratory sounds. First, spectrograms are generated from respiratory sound data using STFT. Then, spectrograms are fed to a CNN and bi-directional LSTM (Bi-LSTM) network for classification. Their approach demonstrated an improved average accuracy of 0.73, which is better than other approaches.

Zhang et al. [95] introduced a frame-level attention model that uses a CNN and bi-directional gated recurrent unit (Bi-GRU) for environmental sound classification using spectro-temporal features and temporal correlations. This model applies an attention mechanism to learn distinguishing feature representations from the sound data. Sound data are first converted to Log Gammatone Spectrograms (Log-GTs). Then input to a CNN. The output of the CNN is fed to a Bi-GRU. This model achieved an accuracy of 93% and 86.1% for ESC-10 and ESC-50 respectively.

Choi et al. [96] proposed a CNN-RNN for classifying music tags. This model also takes advantage of both CNNs and RNNs, enabling effective local feature extraction and temporal summarization of features. Results illustrated that the method provided excellent results in terms of required parameters and training time, demonstrating the potency of the combination of CNNs and RNNs for extracting music features for summarization purposes.

Federico et al. [97] conducted a study to assess how productive an architecture that combines CNNs and LSTMs for general purpose audio event classification and detection using STFT of audio signals. This architecture was examined on DCASE for general purpose audio tagging based on different principles.

Nasrullah and Zhao [98] investigated the use of a temporal architecture for music artist classification. Their work uses STFT signals of music songs and inputs them to a CNN. To capture temporal features, a GRU model is used. Using a CNN-GRU architecture, they applied the work to the artist20 music artist identification dataset by varying audio clip length, dataset split, and feature level. The results indicated that an overall F1-score of 93.7% was obtained over three independent tests, showing the usefulness of their approach.

Feng et al. [99] presented a combined architecture for music genre classification using parallel CNN and Bi-RNN blocks. Spatial features can be extracted using the CNN, and temporal features can be extracted using the Bi-RNN. The outputs of these two blocks are then united to form a significant representation for classification. A parallel network was designed to ensure that the generated features are good representations for music genre classification.

Srivastava et al. [102] proposed the utilization of CNN-GRU and CNN-LSTM for classifying audio signals. MFCCs are used to represent the audio data. The results indicated that the accuracy of the CNN-GRU was 85.7%, while that of the CNN-LSTM was 87.5% on the tested GTZAN.

Nigro et al. [103] evaluated the effectiveness of a Time-Frequency-Energy-emphasis method with respect to Mel spectrograms for acoustic scene classification using a CNN-RNN. Evaluations showed that their method reduced the number of training parameters by half while maintaining better accuracy than using Mel spectrograms.

Qiao et al. [104] introduced a novel sub-spectrogram segmentation method with a CNN-RNN model and score fusion strategy for environmental sound classification. To enhance classification accuracy, their method incorporates score-level fusion method. They conducted experiments to identify the ideal amount and relevant band ranges of sub-spectrograms. Evaluations on ESC-50 revealed an accuracy of 81.9%, which is 9.1% better than the baseline methods.

Jallet et al. [105] applied a CNN-GRU model for acoustic scene classification. Two CNN-GRU models were used; using CNNs as feature extractors and using gated recurrent layers to model the temporal context. Evaluations on DCASE 2017 showed that the accuracy of the two models was 78.9% and 80.8%, respectively, which are higher than the baseline accuracy of 74.8% using a feed-forward NN.

Adavanne et al. [106] introduced a CNN-Bi-LSTM method for sound event classification using low-level spatial features from multi-channel audio signals. To learn multiple types of features, CNNs with Bi-LSTM were used. It was observed that on the TUT-SED 2016 and TUT-SED 2019 datasets, spatial features achieved better F-scores, 6.1% and 2.7%, compared with the monaural features. They suggested that the best approach is to present the features of each channel as separate layers vectors, instead of concatenating them into a single feature vector.

Yang et al. [107] proposed a parallel CNN-Bi-RNN for music genre classification in mobile devices. In the model, two separate blocks (one CNN and one Bi-RNN) extract features in parallel from the STFT spectrograms of the music signals. The extracted features from the CNN and Bi-RNN are then fused to form a vector. The fused vector is then used for the Softmax layer for classification. Evaluations on GTZAN and Extended Ballroom datasets showed that a CNN with a one-layer RNN and a CNN with a two-layers RNN produced the best results compared with other models.

Mounika et al. [108] compared the performance of CNNs and RCNNs for music genre classification. The performance was evaluated using frame acquisition on handmade and GTZAN datasets. They concluded that music information retrieval is challenging as it requires the audio files to be sorted in accordance with their genre.

Gupta et al. [109] proposed a hybrid modeling technique to investigate the performance of various hybrid models for bird species classification. This strategy combined a CNN that uses Mel-frequency and STFT spectrograms as input. The output of the CNNs are then input to different RNNs to combine across time-points. For example, a CNN is combined with LSTM and GRU with Legendre Memory Units (LMU). Results indicated that the hybrid models obtained the most noteworthy accuracy, with an average of 67% and highest accuracy of 90%.

Sang et al. [92] proposed a CNN-LSTM for urban environmental sound classification. The model uses time-domain waveforms as input. CNN was used for feature extraction and the output of the CNN was input to 2-layer LSTM for aggregation of features. They used public UrbanSound8k and their model achieved a classification accuracy of 79.07%. This work also demonstrated that raw waveforms perform better.

Zhang [110] proposed 1D-CNNs with Bi-RNNs for music style classification. To classify music styles, timbre and melody features are generally extracted from music signals. The author claims that the timings are also an important feature for the classification of music style. Therefore, the author first input melody and tonal features to two parallel CNNs. The features extracted from the 1D-CNN models were then input to two Bi-RNNs, and subsequently combined for classification. Experiments on GTZAN demonstrated that their method is effective and obtained an accuracy of 91.99%.

Naranjo-Alcazar et al. [114] presented a convolutional autoencoder model with multi-layer perceptron (MLP). Audio signals are first converted to log-Mel spectrograms. And processed using a convolutional encoder block consisting of three convolution blocks (convolution, batch normalization and ReLU). After obtaining the latent representation, a feature set was input to the convolutional decoder block. The learned latent space representations of the autoencoder were then input to the MLP for classification of audio signals. Both autoencoders (unsupervised) and combined autoencoder and MLP architectures (unsupervised-supervised) settings were experimented, which showed that they can achieve an accuracy of 99%.

Abeßer et al. [117] introduced a smart city monitoring system in urban environments for acoustic scene classification using autoencoder-CNN model. Their system consists of two parts; sensors and server-side. Sensors collect audio recordings from an urban environment using a stacked autoencoder. On the sensor-side, spectrograms patches are extracted, compressed using a denoising autoencoder, and sent to the server-side for classification. For classification, different CNN models are used that were designed to handle dimensionality reduction (that was applied part of encoding). They tested their system to classify five audio classes in real world settings and achieved an accuracy of 75%.

Lin et al. [120] proposed a convolutional capsule autoencoder network (CCAN) to cluster domestic activities from audio recordings. They first converted audio recordings to MFCC spectrograms and input them to an encoder with a convolutional block. The feature embeddings were learned by the decoder and input to a clustering layer. The embeddings were then fed to a decoder with a fully connected layer and deconvolution block. The CCAN was evaluated on DCASE 2018.

Qiu et al. [122] proposed a deep 3D convolutional denoising autoencoder (3D-DCDAE) for music genre classification. Their proposed work is an unsupervised learning model that uses latent music representations. To extract latent representations, unlabeled MIDI files were utilized and input to

3D-DCDAE for denoising and reconstruction. After training, the decoder was replaced with an MLP for classification. They evaluated their autoencoder on the Lakh MIDI dataset.

Another VAE approach was introduced by Latif et al. [124] for speech emotion classification. This is believed to be the first study that applied VAEs for speech emotion classification. They used log-Mel features and VAE-LSTM models for feature representations. Experimental results on the IEMOCAP dataset indicated that features learned using VAEs can achieve superior performance in speech emotion classification compared with other methods.

Asni et al. [125] proposed a convolutional autoencoder for speaker differentiation. They first converted the audio signals into MFCC spectrograms and input them into their convolutional autoencoder. The aim was to classify the audio signals into six categories such as 2-person conversation and 3-person conversation, so forth. They created a custom dataset and evaluated their model. The model achieved an accuracy of 99.29%, 97.62%, 96.43%, 93.43%, and 88.1% for the six categories.

Wilkinghoff and Kurth [126] proposed a deep convolutional autoencoder (DCAE) for acoustic scene classification. They first converted audio signals into log-Mel spectrograms and input them to their model. They conducted experiments on DCASE 2019. The results indicated that on the compared models, their model achieved a significantly higher score of 62.1% on the evaluation dataset of the challenge, which was an improvement from 47.6%.

Irfan et al. [127] introduced a new underwater dataset called deepShip for acoustic scene classification. The dataset consists of real-world underwater sounds of passing by vessels with varying noise levels. In addition, authors proposed a separable convolution autoencoder model for the classification of recordings. Authors investigated various features for the proposed model. Experiments on the proposed dataset showed that the proposed model achieves 77.53% accuracy using CQT feature and outperforms the performance of other methods.

Arniriparian et al. [128] proposed a model by combining a deep convolutional generative adversarial network (DCGAN) with a recurrent sequence to sequence autoencoder (S2SAE) for acoustic scene classification. The features are learned with their model and classification is achieved with an MLP. Authors investigated the effectiveness of their model on DCASE 2017; their model improved the accuracy on the development set to 88.5 % when using only the S2SAE and to 91.1% after fusion.

Another sequence-to-sequence autoencoder toolkit called auDeep was proposed by Freitag et al. [129] for audio processing using combined RNN-autoencoder. It is an open-source toolkit for learning audio representations from spectrograms. The toolkit is freely available to use.

Czyżewski et al. [130] used CNN-autoencoder for unsupervised classification of traffic events. To achieve this, a two-dimensional representation of traffic sounds, created using a one-dimensional convolution layer, is input to an autoencoder then classified with a feed-forward NN.

Gong et al. [137] introduced the first convolution free AST for audio classification, which is a purely attention-based model, and evaluated it on various audio classification benchmarks. They first converted the audio signals into spectrograms then divided the spectrograms into patches and linearly input them to an encoder transformer with position embeddings. They investigated various attention mechanisms and transformers with and without CNN. Their results indicated that AST achieved the best results.

Zeng et al. [144] proposed a method that consists of decision fusion called transformer and causal dilated convolutional network (TCDCN) for audio event recognition. They first converted audio signals into Mel-spectrogram images and input them to the dilated CNN for feature extraction. They then input the extracted features to an attention module for classification. For the audio recognition task, they cropped data from YouTube audio clips to form a custom dataset. The results indicated that the TCDCN outperformed NNs and other fusion models.

2) HYBRID DEEP LEARNING MODELS WITH TRADITIONAL CLASSIFIERS

In this section, we discuss hybrid deep learning models with traditional classifiers for audio classification. In the literature, we found cases of hybrid models that combine different deep learning models with traditional machine learning methods to create new and more powerful models. The aim with hybrid models is to combine the strengths of deep learning models with those of traditional machine-learning methods such as those using SVMs and KNN [160], as shown in Figure 8.

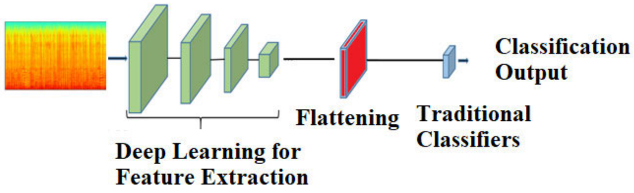


FIGURE 8. Hybrid models for audio classification.

Hybrid models with traditional classifiers generally use deep learning models such as CNNs and RNNs for feature extraction. Instead of classification in an end-to-end network, the extracted features are then flattened and input to traditional classifiers. Hybrid models are used in different applications such as audio classification, NLP, image recognition, and autonomous driving.

a: DEEP LEARNING METHODS THAT USE HYBRID MODELS WITH TRADITIONAL CLASSIFIERS

Demir et al. [153] proposed a pyramidal concatenated CNN for environmental sound classification. Sound signals are first converted to spectrogram images using STFT. Then, various deep learning models are used for feature extraction,

TABLE 5. Comparison of hybrid deep learning models.

Reference /Years	Method	Input Data/feature sets	Deep Learning Architecture	Dataset	Performance	Application
[92] 2018	Hybrid	Raw waveform	CNN-LSTM	UrbanSound8k dataset	79%	Urban sound classification
[93] 2020	Hybrid	Time domain in-phase	CNN-LSTM	RadioML2016	92.1%	Music classification
[94] 2021	Hybrid	spectrogram	CNN- Bi-LSTM	ICBHI2017 challenge respiratory sound database	73%	Respiratory sound classification
[95] 2020	Hybrid	Log-gammatone spectrogram (Log-GTs)	Attention CNN-Bi-GRU	ESC-50 and ESC-10 datasets	93.7%	Environmental sound classification
[96] 2017	Hybrid	Log-amplitude Mel-spectrogram	CNN-RNN	Million Song Dataset		Music tagging classification
[97] 2018	Hybrid	STFT	CNN-LSTM	DCASE 2018 task 2 dataset	77,98%	Audio events classification
[98] 2019	Hybrid	STFT	CNN-GRU	Artist20 dataset	93.7%	Music artists classification
[99] 2017	Hybrid	(STFT) spectrogram	Parallel CNN-GRU	GTZAN dataset	92%	Music genre classification
[102] 2022	Hybrid	MFCCs	CNN-GRU CNN-LSTM	GTZAN dataset	87.5%	Audio classification
[103] 2022	Hybrid	Mel-spectrogram	CNN-RNN			Acoustic scene classification
[104] 2019	Hybrid	Sub-spectrogram segmentation	CNN-RNN	ESC-50	81.9%	Environmental sound classification
[105] 2017	Hybrid	Small shift-invariant features es from a time-frequency representation	CNN-GRU	DCASE 2017	80.8%	Acoustic scene classification
[106] 2017	Hybrid	Log-Mel band energies (Mel-monaural)	CNN-Bi-LSTM	TUT-SED 2016 and TUT-SED 2009	84.6%	Sound event detection
[107] 2020	Hybrid	STFT	Parallel CNN-Bi-RNN	GTZAN and Extended Ballroom	92.5%	Music genre classification
[108] 2021	Hybrid	Mel-spectrogram	CNN-RNN	GTZAN and handmade	73.2%	Music genre classification
[109] 2021	Hybrid	Mel-spectrogram and STFT	CNN-LSTM-GRU-LMU	Cornell Bird Challenge (CBC)2020 dataset	90%	Bird species classification
[92] 2018	Hybrid	Raw waveforms	CNN-LSTM	UrbanSound8k	79.06%	Environmental sound classification
[110] 2021	Hybrid	1D music signal	CNN-Bi-RNN	GTZAN data	91.99%	Music style classification
[114] 2020	Autoencoder	Log-Mel spectrogram	Autoencoder with MLP	Dataset for FSL and OSR	99%	Audio classification
[117] 2017	Hybrid	STFT	Autoencoder-CNN	TUT Sound Events, e Urban Sound Dataset, IEEE AASP public & private datasets, YouTube	75%	Acoustic scene classification
[120] 2021	Hybrid	MFCC	CNN-autoencoder	DCASE 2018	61.91%	Domestic activity clustering
[122] 2021	Hybrid	MIDI2Img	CNN-autoencoder	Lakh MIDI dataset	88.30%	Music genre classification
[124] 2017	Hybrid	Log-Mel speech frame representation	Variational autoencoders VAE-LSTM	IEMOCAP) [25] dataset	64.93%	Speech emotion classification
[125] 2018	Hybrid	MFCC	Convolutional autoencoder	Custom and Mozilla's Common Voice dataset	<90%	Speaker differentiation classification
[126] 2019	Hybrid	Log-Mel spectrogram	Deep convolutional autoencoder	(DCASE) Challenge 2019	62%	Acoustic scene classification
[127] 2021	Hybrid	Cepstrum Mel, MFCC, CQT, GFCC, Wavelet packets	Separable convolutional autoencoder	DeepShip	77.53%	Underwater audio classification

TABLE 5. ***(Continued.)*** **Comparison of hybrid deep learning models.**

[128] 2018	Hybrid	Mel-spectrogram	Combined DCGAN with S2SAE	DCASE 2017	91%	Acoustic scene classification
[129] 2017	Hybrid	Spectrogram	RNN-Autoencoder	TUT AS, ESC−10, ESC−50, GTZAN	88%	Audio processing
[130] 2019	Hybrid	Audio signal	CNN-Autoencoder	custom	87%	Traffic event classification
[137] 2021	Hybrid	Spectrogram	CNN-Audio Spectrogram Transformers	ESC-50, Speech Command V2	98.1%	Environmental sound classification and speech command classification
[144] 2021	Hybrid	Mel-spectrogram	Dilated CNN-Transformer	Audio clips cropped by YouTube	91%	Audio event recognition

TABLE 6. **Comparison of hybrid deep learning models with traditional classifiers.**

Reference /Years	Method	Input Data/feature sets	Deep Learning Architecture	Dataset	Performance	Application
[153] 2020	Hybrid	STFT	CNN (VGG16, VGG19 and DenseNet201) +SVM	ESC-10, ESC-50 and UrbanSound8K	94.8%, 81.4%, 78.14%	Environmental sound classification
[154] 2017	Hybrid	MFCC	RNN (GRU) with SVM as a classifier	LITIS Rouen dataset	97.7%	Audio scene classification
[155] 2018	Hybrid Dilated CNN	Mel-frequency cepstral coefficients	Hybrid: CNN+different classifiers	GTZAN dataset	91%	Music genre classification
[55] 2023	Hybrid	Mel-spectrogram	Autoencoder with RF and SVM	Bird species dataset (Not specified)	94%	Audio classification
[156] 2019	Hybrid	Continuous wavelet transform	Autoencoder-SVM	R.A.L.E. Lung Sounds	86.51%	Lung sound classification
[157] 2020	Hybrid	STFT	CNN with different classifiers (SVM, KNN, decision trees, etc.)	ESC-10	95.8%	Environmental sound classification
[158] 2020	Hybrid	Mel-spectrogram	CNN with Broad Learning	GTZAN, Ballroom, and Emotion)	95%	Music Genre classification
[159] 2018	Hybrid	SSD, MFFC, Spectral roll off, Zero crossing rate, Chroma frequency, Rhythm Histograms, Spectral centroid	LSTM with SVM	GTZAN	89%	Music genre classification

such as DesNet201, VGG16, and VGG19. Since during the feature extraction process, the feature vector is quite large, a pyramidal approach is used by using feature concatenation and reduction. Therefore, feature vector size decreases. Instead of classification in an end-to-end deep learning network, the obtained features are input to an SVM for classification. On ESC-10, ESC-50, and UrbanSound8K, their model achieved an accuracy of 94.8%, 81.4% and 78.14%, respectively.

Phan et al. [154] developed a powerful technique for scene classification from audio using DRNNs. The audio scene is first converted into a series of label tree embedding feature vectors then segmented into various sections. To classify the subsequences, a deep GRU based RNN is used. To obtain the overall label for the whole sequence, the outputs of the segment classifications are combined. The output is then input to a linear SVM for classification. Therefore, this is a hybrid model that extracts features using an RNN but classification is achieved using a traditional classifier, e.g., an SVM. The LITIS Rouen dataset tests showed that the hybrid model of RNN-SVM achieved an F1-score of 97.7%.

Li et al. [155] investigated a dilated CNN for music genre classification that combines a CNN with different traditional classifiers such as an SVM, RF and Gaussian Discriminant Analysis (GDA). They used spectrograms that use MFCCs as input to the dilated CNN. In the dilated

CNN, convolution filters are applied with a gap between them, hence dilating the convolution operation and extracted features. After training the dilated CNN model, they input the extracted features to different machine learning classifiers for classification. Experiments on GTZAN demonstrated that their hybrid model improved the classification accuracy and achieved 91%. It also performed better than the dilated CNN only.

Vamsi et al. [55] proposed an autoencoder-based hybrid model for bird species classification from sound recordings. They first applied pre-processing to identify pitch in the audio signal then using the frequency waves, converted the pre-processed audio signals into Mel frequency spectrograms and input them to an autoencoder for feature extraction. Finally, classification was achieved using the traditional classifiers RF and SVM; the extracted features were input to RF and the SVM for classification. Experiments on a bird species dataset showed that their hybrid model was effective.

Falah and Jondri [156] proposed a hybrid model that combines stacked autoencoders with an SVM for lung sound classification. They used the continuous wavelet transform of lung sounds as input to stacked autoencoders for feature selection. They then used the SVM for classification. Additionally, discrete wavelet transform was also compared. The results indicated that their hybrid model with the continuous wavelet transform achieved 86.51% accuracy.

Ullo et al. [157] developed a hybrid model for environmental sound classification using a CNN and different traditional classifiers. They first converted audio signals into spectrograms using STFT. Then applied feature extraction from spectrograms using two pre-trained models such as AlexNet and VGG-16. For classification, the extracted features were input to various classifiers; decision trees, SVMs, KNN, and softmax. Experiments on ESC-10 demonstrated that the CNN with softmax layer and CNN with KNN achieved the best results.

Tang and Chen [158] proposed a hybrid model for music genre classification by combining an RCNN with a broad learning (BL) technique. They first converted audio signals into Mel-spectrograms then, using the RCNN, carried out feature extraction. Finally, for the classification, Broad Learning (BL technique receives the extracted features and predicts the music genre. Experiments on GTZAN, Ballroom, and Emotion, showing that their hybrid model is effective.

Fulzele et al. [159] also presented a hybrid model that is a combination of LSTM and an SVM for music genre classification. They utilized LSTM for feature extraction and the SVM as a classifier on GTZAN. Evaluations indicated that their hybrid model performed better than individual performances of LSTM and the SVM and achieved an accuracy of 89%.

III. DATASETS

An audio dataset is a collection of audio recordings that can be used to train a deep learning model to recognize and classify different types of audio content/signals. We collected information on several audio datasets commonly used by various researchers for audio classification.

One of the popular datasets that is used for environmental sound classification is ESC-50 [162]. This dataset consists of 2000 environmental audio recordings organized into 50 semantically-uniform classes of common sound events, such as Dog, Rain, Sea waves, and Thunderstorm, with each class containing forty 5-second recordings. As Figure 9 illustrates, the dataset provides a comprehensive taxonomy of sounds related to the forest environment. Similarly, ESC-10 is a subset of ESC-50 dataset's taxonomy [163], and composed of ten classes, representing three general sound groups: 1) transient/percussive sounds with meaningful temporal patterns (sneezing, dog barking, clock ticking), 2) sound events with harmonic content (crying baby, crowing rooster), and 3) structured noise/soundscapes (rain, sea waves, fire crackling, helicopter, chainsaw).

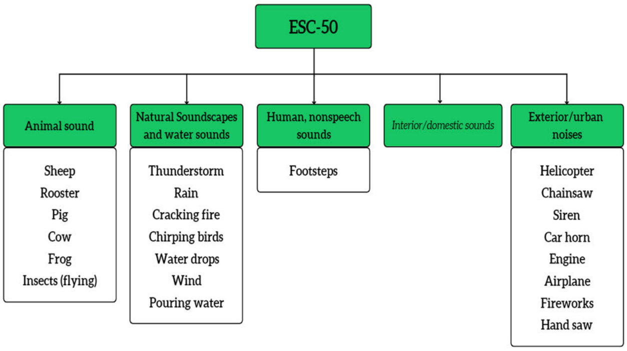


FIGURE 9. ESC-50 dataset [162].

UrbanSound8k [164] was designed for urban sound classification and sound event detection. It is a well-known and widely used dataset. It consists of 8732 labeled audio samples of urban sounds from 10 classes and falling into different sound categories, as shown in Figure 10.

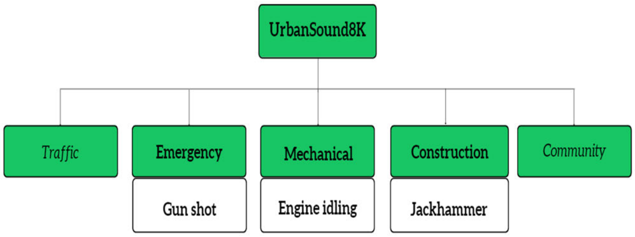


FIGURE 10. Urbansound8K dataset [162], [164].

GTZAN is another common dataset used for music genre classification. It includes audio tracks from ten music genres: Rock, Pop, Country, Blues, Jazz, Latin, Reggae, Classical, Hip-Hop, and Metal [165]. This dataset can be used for both supervised and unsupervised learning tasks, enabling exploration of genre relationships, comparison of genre trends, and identification of similarities between different genres. It contains over 7,500 audio tracks.

The YouTube-100M audio dataset is a comprehensive collection of audio files from YouTube videos, with over 100 million tracks spanning a range of genres and topics [67]. It contains audio in a variety of formats, including MP3, WAV, and FLAC, and organized into a hierarchical structure by genre, artist, and album. Metadata such as artist name, release date, and track length, are also provided.

The Mozilla Common Voice dataset is a dataset for various speaker classification tasks including speech recognition, speaker recognition, and language identification [166]. It includes audio samples that are labeled as having background noise or silence.

The VoxCeleb dataset contains an extensive collection of over 100,000 utterances from 1,251 different celebrities [167]. While designing the dataset, gender-balance was considered; 55% male speakers and 45% female speakers. The dataset is composed of speakers of various ethnicities, accents, professions and ages.

The EmoDB dataset is a collection of emotion-annotated audio recordings of German-speaking actors for emotion recognition from speech [168]. It consists of over 600 sentences spoken by 10 actors in 4 different emotional states (anger, boredom, happiness and sadness). Each sentence is labeled with one of the four emotion states. The corpus also contains additional information such as gender, age, and dialect of the speaker.

The MUSAN dataset consists of 109 hours of music, speech, and noise audio recordings [169]. It is broken down into speech, music, and noise folders and organized by the source website from which the recording was downloaded. All audio files are available as 16-kHz WAV files.

The Respiratory Sound Database [170] is a collection of audio samples acquired from research in hospitals in Portugal and Greece for disease classification. The database contains a total of 5.5 hours of recordings from 126 subjects, including 6898 respiratory cycles, 1864 of which containing crackles, 886 containing wheezes, and 506 containing both.

For acoustic scene classification and event detection, the TUT Sound Events 2017 [171] and TUT Sound Events 2016 [172] datasets are widely used. TUT Sound Events 2017 was generated in a street environment and consists of 24 audio recordings in 6 different classes: People speaking, people walking, brakes squeaking, car, children, and large vehicle. TUT Acoustic Scenes 2016 was designed for sound event detection and is a collection of binaural recordings in 15 different acoustic environments. It includes recordings in residential areas and home environments and manually annotated with the label, onset, and offset of sound events.

AudioSet is a collection of 10-second sound clips obtained from YouTube videos labeled by human annotators. To acquire the data, YouTube content was searched, and segments were created for annotation, which was then verified by human annotators. The dataset contains a wide variety of sounds such as musical instruments, human speech, animals, and environmental sounds.

The Speech Commands dataset [173] contains spoken words and designed to assist keyword spotting systems. Keyword spotting is a task to identify a single spoken word among other words with high accuracy.

The DCASE dataset is an audio dataset created to support research in sound event detection and classification [174]. First launched by Queen Mary University of London in 2013, the DCASE dataset encompasses a wide range of acoustic scenes and events, from environmental, urban, and domestic sounds, and includes a variety of real-world scenarios and recording conditions. Thus far, the dataset has seen several editions, such as DCASE 2013, DCASE 2016, DCASE 2017, DCASE 2020, DCASE 2021, and DCASE 2022.

IV. DISCUSSION OFDEEP LEARNING-BASED METHODS

Audio classification using deep learning models, including CNNs, RNNs, autoencoders, transformers, and hybrid models (hybrid deep learning models and hybrid deep learning models with traditional classifiers), has emerged as a promising approach for analyzing and classifying audio data. We first explained (a) the main components of different deep learning architectures for audio classification and then discussed (b) the deep learning-based methods that apply these models. In this section, we highlight the important features of these different deep learning models in terms of method, input data/feature, architectures, datasets, model performance, and model application. Figure 11 shows the distribution of the reviewed methods using different deep learning models. It was observed that a CNN was the most preferred deep learning architecture. Autoencoders, transformers and hybrid models such as CNN-LSTM, CNN-Bi-RNN, CNN-autoencoders are also preferred. Although transformers are relatively new compared with other deep learning architectures, it is expected that more transformer-based methods will emerge in future.

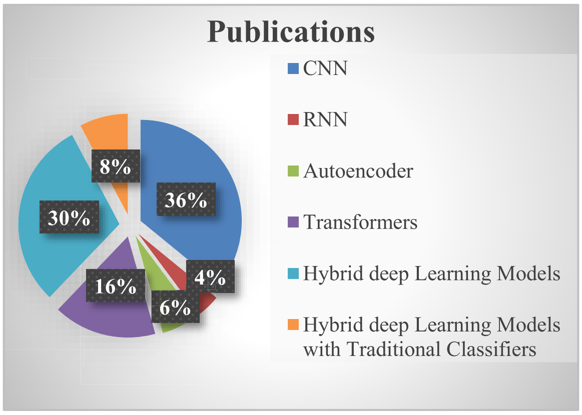


FIGURE 11. Number of publications for each type of architecture.

The following are our observations according to the structured review of different CNN-based methods for audio classification. CNN-based methods have shown to have high performance in different applications such as classification of

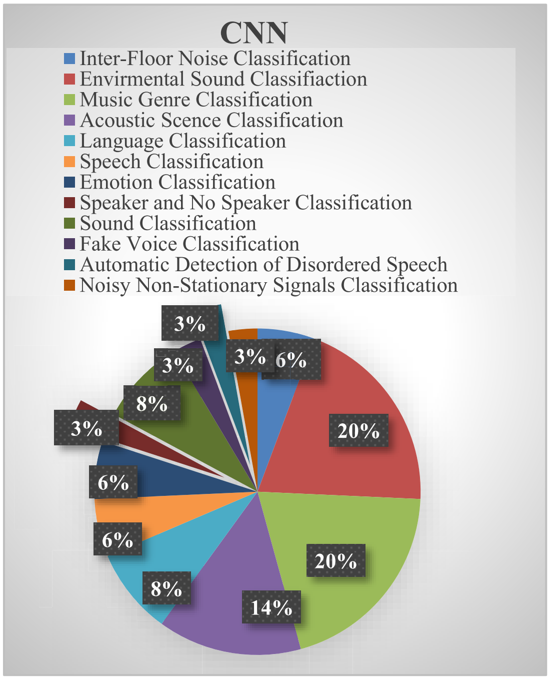


FIGURE 12. Application areas of CNNs.

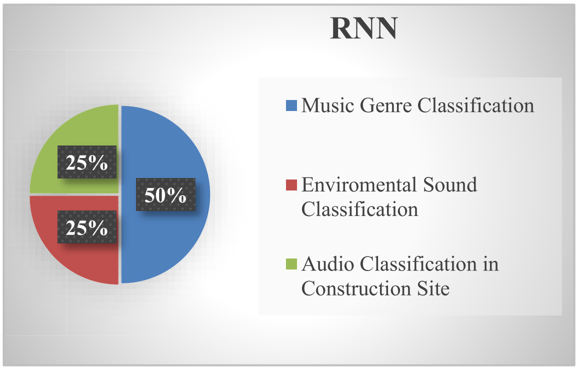


FIGURE 13. Application areas of RNNs.

inter-floor noise, environmental noise, environmental sounds, soundtrack music genre, English accent, emotion, acoustic scene, and speech classification, as shown in Table 1. Common audio classification application areas of CNN-based methods are illustrated as a pie chart in Figure 12. Environmental sound classification (22%), music genre classification (21%) and acoustic scene classification (15%) are the top three application areas. However, different researchers have used different feature extraction techniques for CNNs, but the most common ones are using Mel spectrograms, MFCCs, and STFT. Different techniques are generally used for CNN architectures; such as custom CNNs and some methods use common CNNs such as VGG16 and DenseNet.

Beyond CNN-based methods for audio classification, hybrid CNN models with other deep learning architectures have advanced significantly in the audio classification domain. In particular, many RNN, autoencoder and transformer-based methods utilize CNN to form a hybrid deep learning model.

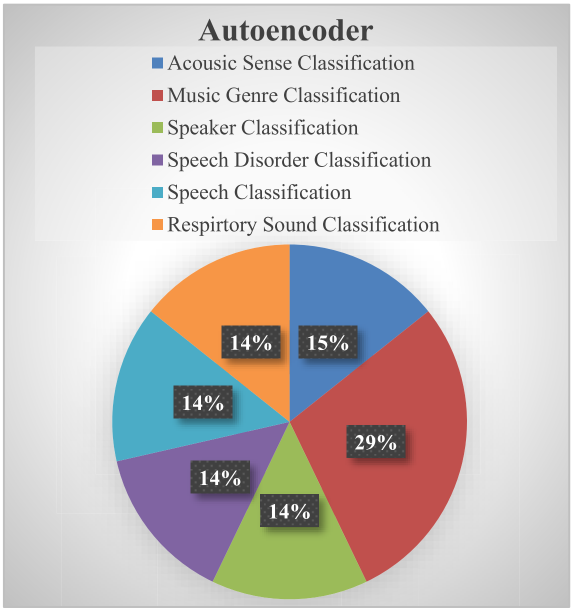


FIGURE 14. Application areas of autoencoders.

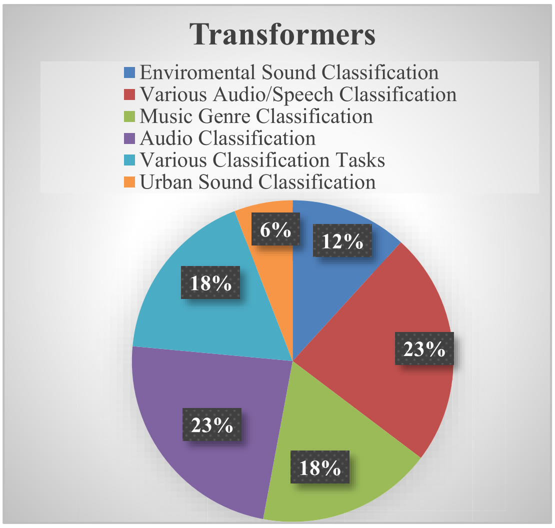


FIGURE 15. Application areas of transformers.

As mentioned above and illustrated in Table 5, only few standalone RNN methods are used for audio classification. Many methods do not use RNNs alone but combine them with CNNs such as convolutional and bi-directional models. In terms of application areas of RNN-based methods, music genre classification (45%) is the most common.

Comparative analysis of methods based on autoencoders is shown in Table 3. It has been observed that different researchers used autoencoders to classify audio signals using various datasets and feature extraction techniques. The most commonly used autoencoder architectures include VAEs, and autoencoders with an MLP. In terms of application, acoustic

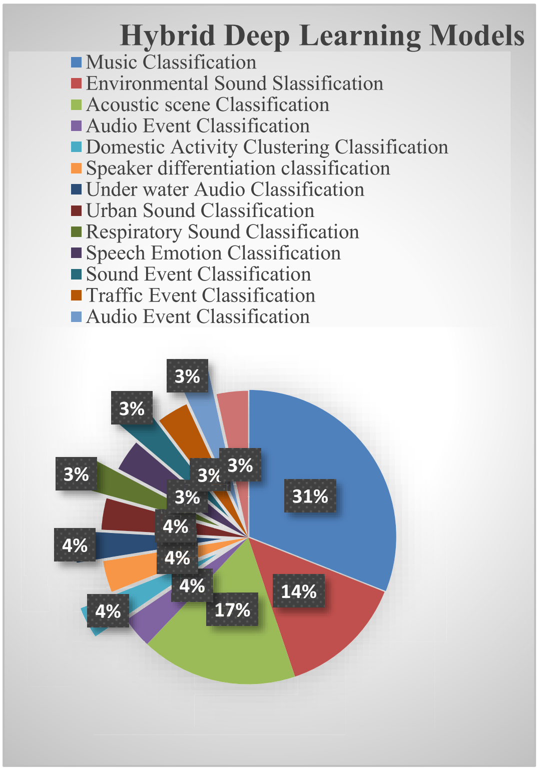


FIGURE 16. Application areas of hybrid deep learning models.

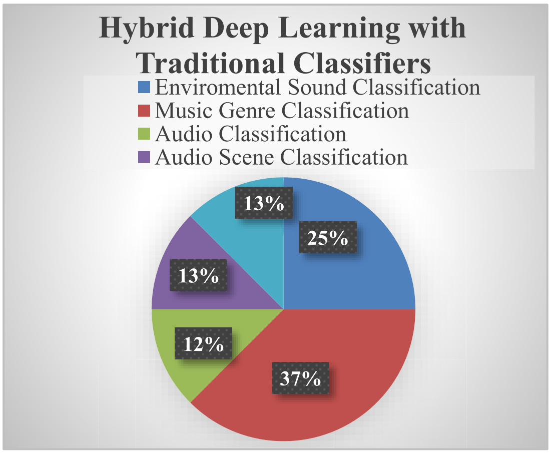


FIGURE 17. Application areas of hybrid deep learning models with traditional classifiers.

scene classification (15%) and music genre classification (29%) are common for autoencoders, as shown in Figure 14.

The comparative analysis of transformer-based methods for audio classification is summarized in Table 4. It has been observed that different transformers are applied to audio data such as ASTs and PaSSTs, and other types of transformers have been emerging for various audio classification tasks. As a future extraction generally, spectrograms are used. Transformer-based methods are generally used for music genre classification, environmental/acoustic scene classification and various other tasks, as shown in Figure 15.

Comparison of hybrid models for audio classification is shown in Table 5 and 6. Hybrid models that combine the strengths of deep learning with other different deep learning models or traditional classifiers. Hybrid models are used for different application areas; music genre classification and environmental/acoustic scene classification are the most common, as shown in Figure 16 and 17.

V. CONCLUSION AND FUTURE DIRECTIONS

We mainly focused on audio classification while using different deep learning models such as CNNs, RNNs, autoencoders, transformers, and hybrid models. We provided a comprehensive overview of the recent advancements for audio classification using deep learning. To implement a deep learning model, the first step is to convert the 1D audio signal to a 2D spectrogram. This can be achieved using feature extraction techniques such as using STFT, Mel-spectrograms, Log-Mel spectrograms, and other time-frequency representations. An appropriate deep learning model can then be used to classify audio signals. We examined the research on audio classification, focusing on input feature extraction, deep learning architectures, datasets, performance, and application area. This review suggests that deep learning models are powerful tools for audio classification. These models produced promising results in various audio classification applications, including speech classification, music genre classification, environmental sound classification, noise classification, acoustic scene classification, and speech-emotion classification.

CNNs have proven to be highly effective for extracting spatial features from audio signals, making them suitable for tasks such as music genre classification and environmental sound classification. The remarkable success of CNNs in this field is due to their ability to capture high-level features from the audio data. RNNs are particularly well-suited for tasks that require temporal dependencies, such as speech classification and audio-sequence classification. This is due to their ability to capture time-dependent information in the data. Autoencoders can be used for unsupervised learning of features and reducing the dimensions of audio data. By reconstructing the input signal, these models can learn to identify and capture the important characteristics of the audio data, enabling accurate classification. Transformer-based methods, specifically spectrogram- and MFCC-based representations, have become common for audio classification tasks due to their ability to provide a concise and meaningful representation of audio signals. This representation enables efficient model training and accurate classification. This survey highlighted the potential of hybrid models for audio classification that combine different deep learning models with traditional classifiers. Hybrid models leverage the strengths of multiple architectures, enabling more comprehensive feature extraction and capturing both spatial and temporal dependencies in audio data.

Regarding feature extraction, we highlighted and summarized the significance of spectrogram-based representations such as STFT, Mel-spectrograms, Log-Mel spectrograms, MFCCs, and other time-frequency representations, as effective features for audio classification. These features encode important spectral and temporal information, enabling deep learning models to learn discriminative representations.

This survey also highlighted the significance of datasets with high-quality, diverse labels for training and evaluating audio classification models. The emergence of large-scale datssets, such as UrbanSound8, GTZAN, DCASE, ESC-50, Urban Sound, Common voice Mozilla and Audio Set, has enabled progress in audio classification research and benchmarking. Nevertheless, there are still challenges that need to be addressed, including class imbalance, limited domain-specific datasets, and privacy issues related to audio data.

In summary, this survey focused on deep learning models, including CNNs, RNNs, autoencoders, transformers, and hybrid models, which hold great promise for audio classification tasks. We expect to see more transformer-based methods. Many transformer-based methods have been proposed that are combined with other deep learning models. In future, we expect to see more hybrid models using transformers. The findings of this survey can guide researchers in selecting appropriate methods, architectures and datasets as well as inspire future research directions to address the challenges and advance audio classification while using deep learning models.

REFERENCES

[1] B. Kaur and J. Singh. (2021). *Audio Classification: Environmental Sounds Classification*. [Online]. Available: https://hal.science/hal-03501143/document

[2] G. Guo and S. Z. Li, "Content-based audio classification and retrieval using SVM learning," in *Proc. 1st IEEE Pacific-Rim Conf. Multimedia*, Dec. 2000, pp. 1–4. [Online]. Available: https://www.microsoft.com/en-us/research/wpcontent/uploads/2016/02/content_audio_classification.pdf

[3] G. Guo and S. Z. Li, "Content-based audio classification and retrieval by support vector machines," *IEEE Trans. Neural Netw.*, vol. 14, no. 1, pp. 209–215, Jan. 2003, doi: 10.1109/TNN.2002.806626.

[4] L. Lu, H.-J. Zhang, and S. Z. Li, "Content-based audio classification and segmentation by using support vector machines," *Multimedia Syst.*, vol. 8, no. 6, pp. 482–492, Apr. 2003, doi: 10.1007/s00530-002-0065-0.

[5] L. Chen, S. Gunduz, and M. Ozsu, "Mixed type audio classification with support vector machine," in *Proc. IEEE Int. Conf. Multimedia Expo*, Jul. 2006, pp. 781–784.

[6] Y. Zhu, Z. Ming, and Q. Huang, "SVM-based audio classification for content-based multimedia retrieval," in *Multimedia Content Analysis and Mining*. Cham, Switzerland: Springer, 2007, pp. 474–482.

[7] P. Dhanalakshmi, S. Palanivel, and V. Ramalingam, "Classification of audio signals using SVM and RBFNN," *Expert Syst. Appl.*, vol. 36, no. 3, pp. 6069–6075, Apr. 2009.

[8] V. Elaiyaraja and P.M. Sundaram, "Audio classification using support vector machines and independent component analysis," *J. Comput. Appl.*, vol. 5, no.1, pp. 34–38, 2012.

[9] L. Bahatti, O. Bouattane, M. Elhoussine, and M. Hicham, "An efficient audio classification approach based on support vector machines," *Int. J. Adv. Comput. Sci. Appl.*, vol. 7, no. 5, pp. 1–7, 2016.

[10] S. Souli and Z. Lachiri, "Audio sounds classification using scattering features and support vectors machines for medical surveillance," *Appl. Acoust.*, vol. 130, pp. 270–282, Jan. 2018.

[11] L. Lu, H.-J. Zhang, and H. Jiang, "Content analysis for audio classification and segmentation," *IEEE Trans. Speech Audio Process.*, vol. 10, no. 7, pp. 504–516, Oct. 2002.

[12] T. L. Priya, N. R. Raajan, N. Raju, P. Preethi, and S. Mathini, "Speech and non-speech identification and classification using KNN algorithm," *Proc. Eng.*, vol. 38, pp. 952–958, Jan. 2012.

[13] S. Zahid, F. Hussain, M. Rashid, M. H. Yousaf, and H. A. Habib, "Optimized audio classification and segmentation algorithm by using ensemble methods," *Math. Problems Eng.*, vol. 2015, Jan. 2015, Art. no. 209814.

[14] R. Thiruvengatanadhan, "Speech/music classification using MFCC and KNN," *Int. J. Comput. Intell. Res.*, vol. 13, no. 10, pp. 2449–2452, 2017.

[15] M. Murugappan, "Human emotion classification using wavelet transform and KNN," in *Proc. Int. Conf. Pattern Anal. Intell. Robot.*, vol. 1, Jun. 2011, pp. 148–153.

[16] X. Shao, C. Xu, and M. S. Kankanhalli, "Applying neural network on the content-based audio classification," in *Proc. 4th Int. Conf. Inf., Commun. Signal Process., 4th Pacific Rim Conf. Multimedia.*, vol. 3, 2003, pp. 1821–1825.

[17] H. Meinedo and J. Neto, "A stream-based audio segmentation, classification and clustering pre-processing system for broadcast news using ANN models," in *Proc. 9th Eur. Conf. Speech Commun. Technol.*, Sep. 2005, pp. 1–4.

[18] C. Freeman, R. D. Dony, and S. M. Areibi, "Audio environment classication for hearing aids using artificial neural networks with windowed input," in *Proc. IEEE Symp. Comput. Intell. Image Signal Process.*, Apr. 2007, pp. 183–188.

[19] V. Mitra and C.-J. Wang, "Content based audio classification: A neural network approach," *Soft Comput.*, vol. 12, no. 7, pp. 639–646, Oct. 2007.

[20] K. Karthikeyan, D. R. Mala, "Content based audio classifier feature extraction using ANN techniques," *Int. J. Innov. Res. Adv. Eng.*, vol. 5, no. 5, Apr. 2018.

[21] C.-Y.-J. Peng, K. L. Lee, and G. M. Ingersoll, "An introduction to logistic regression analysis and reporting," *J. Educ. Res.*, vol. 96, no. 1, pp. 3–14, Sep. 2002.

[22] Z. Fu, G. Lu, K. M. Ting, and D. Zhang, "Learning naive Bayes classifiers for music classification and retrieval," in *Proc. 20th Int. Conf. Pattern Recognit.*, Aug. 2010, pp. 4589–4592.

[23] Y. An, S. Sun, and S. Wang, "Naive Bayes classifiers for music emotion classification based on lyrics," in *Proc. IEEE/ACIS 16th Int. Conf. Comput. Inf. Sci. (ICIS)*, May 2017, pp. 635–638.

[24] C. Kraetzer, A. Oermann, J. Dittmann, and A. Lang, "Digital audio forensics: A first practical evaluation on microphone and environment classification," in *Proc. 9th workshop Multimedia Secur.*, 2007, pp. 63–74.

[25] S. K. Bhakre and A. Bang, "Emotion recognition on the basis of audio signal using naive Bayes classifier," in *Proc. Int. Conf. Adv. Comput., Commun. Informat. (ICACCI)*, Sep. 2016, pp. 2363–2367.

[26] Z. Liu, J. Huang, and Y. Wang, "Classification TV programs based on audio information using hidden Markov model," in *Proc. IEEE 2nd Workshop Multimedia Signal Process.*, Dec. 1998, pp. 27–32.

[27] X. Shao, C. Xu, and M. S. Kankanhalli, "Unsupervised classification of music genre using hidden Markov model," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, Jun. 2004, pp. 2023–2026, doi: 10.1109/icme.2004.1394661.

[28] H. Meng and N. Bianchi-Berthouze, "Naturalistic affective expression classification by a multi-stage approach based on hidden Markov models," in *Proc. 4th Int. Conf., Affect. Comput. Intell. Interact. (ACII)*, Memphis, TN, USA., Jan. 2011, pp. 378–387, doi: 10.1007/978-3-642-24571-8_49.

[29] A. Pikrakis, S. Theodoridis, and D. Kamarotos, "Classification of musical patterns using variable duration hidden Markov models," *IEEE Trans. Audio, Speech Lang. Process.*, vol. 14, no. 5, pp. 1795–1807, Sep. 2006, doi: 10.1109/TSA.2005.858542.

[30] L. Jian, "Automatic audio classification by using hidden Markov model," *J. Softw.*, vol. 13, no. 8, pp. 1593–1597, Jan. 2002.

[31] P. Ahrendt, J. Larsen, and C. Goutte, "Co-occurrence models in music genre classification," in *Proc. IEEE Workshop Mach. Learn. Signal Process.*, Sep. 2005, pp. 247–252, doi: 10.1109/MLSP.2005.1532908.

[32] A. Flexer, "A closer look on artist filters for musical genre classification," *World*, vol. 19, no. 122, pp. 16–17, 2007. [Online]. Available: http://ismir2007.ismir.net/proceedings/ISMIR2007_p341_flexer.pdf

[33] H. Lu, H. Zhang, and A. Nayak, "A deep neural network for audio classification with a classifier attention mechanism," 2020, *arXiv:2006.09815*.

[34] Y. Cui and F. Wang, "Research on audio recognition based on the deep neural network in music teaching," *Comput. Intell. Neurosci.*, vol. 2022, May 2022, Art. no. 7055624, doi: 10.1155/2022/7055624.

[35] 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, doi: 10.3390/jsan10040072.

[36] A. Abeysinghe, S. Tohmuang, J. L. Davy, and M. Fard, "Data augmentation on convolutional neural networks to classify mechanical noise," *Appl. Acoust.*, vol. 203, Feb. 2023, Art. no. 109209, doi: 10.1016/j.apacoust.2023.109209.

[37] X. Dong, B. Yin, Y. Cong, Z. Du, and X. Huang, "Environment sound event classification with a two-stream convolutional neural network," *IEEE Access*, vol. 8, pp. 125714–125721, 2020, doi: 10.1109/ACCESS.2020.3007906.

[38] G. Bahle, V. Fortes Rey, S. Bian, H. Bello, and P. Lukowicz, "Using privacy respecting sound analysis to improve Bluetooth based proximity detection for COVID-19 exposure tracing and social distancing," *Sensors*, vol. 21, no. 16, p. 5604, Aug. 2021, doi: 10.3390/s21165604.

[39] GeeksforGeeks. (Jan. 2023). *Advantages and Disadvantages of Deep Learning*. [Online]. Available: https://www.geeksforgeeks.org/advantages-and-disadvantages-of-deep-learning/

[40] S. Akinpelu and S. Viriri, "Speech emotion classification: A survey of the state-of-the-art," in *Proc. 2nd EAI Int. Conf., Pan-African Artif. Intell. Smart Syst. (PAAISS)*. Cham, Switzerland: Springer, Feb. 2023, pp. 379–394. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-031-25271-6_24

[41] C. V. Reshma and R. Rajasree, "A survey on speech emotion recognition," in *Proc. IEEE Int. Conf. Innov. Commun., Comput. Instrum. (ICCI)*, Mar. 2019, pp. 193–195, doi: 10.1109/ICCI46240.2019.9404432.

[42] A. Dandashi and J. M. AlJaam, "A survey on audio content-based classification," in *Proc. Int. Conf. Comput. Sci. Comput. Intell. (CSCI)*, Dec. 2017, pp. 408–413, doi: 10.1109/csci.2017.69.

[43] Z. Fu, G. Lu, K. M. Ting, and D. Zhang, "A survey of audio-based music classification and annotation," *IEEE Trans. Multimedia*, vol. 13, no. 2, pp. 303–319, Apr. 2011, doi: 10.1109/TMM.2010.2098858.

[44] D. C. Corrêa and F. A. Rodrigues, "A survey on symbolic data-based music genre classification," *Expert Syst. Appl.*, vol. 60, pp. 190–210, Oct. 2016, doi: 10.1016/j.eswa.2016.04.008.

[45] A. Bansal and N. K. Garg, "Environmental sound classification: A descriptive review of the literature," *Intell. Syst. Appl.*, vol. 16, Nov. 2022, Art. no. 200115, doi: 10.1016/j.iswa.2022.200115.

[46] K. B. Bhangale and M. Kothandaraman, "Survey of deep learning paradigms for speech processing," *Wireless Pers. Commun.*, vol. 125, no. 2, pp. 1913–1949, Mar. 2022, doi: 10.1007/s11277-022-09640-y.

[47] V. Roger, J. Farinas, and J. Pinquier, "Deep neural networks for automatic speech processing: A survey from large corpora to limited data," *EURASIP J. Audio, Speech, Music Process.*, vol. 2022, no. 1, p. 19, Aug. 2022, doi: 10.1186/s13636-022-00251-w.

[48] J. Abeßer, "A review of deep learning based methods for acoustic scene classification," *Appl. Sci.*, vol. 10, no. 6, p. 2020, Mar. 2020, doi: 10.3390/app10062020.

[49] A. Khan, A. Sohail, U. Zahoora, and A. S. Qureshi, "A survey of the recent architectures of deep convolutional neural networks," *Artif. Intell. Rev.*, vol. 53, no. 8, pp. 5455–5516, Apr. 2020, doi: 10.1007/s10462-020-09825-6.

[50] S. Sukittanon, A. C. Surendran, J. C. Platt, and C. J. C. Burges, "Convolutional networks for speech detection," in *Proc. Interspeech*, Oct. 2004, pp. 1–4, doi: 10.21437/interspeech.2004-376.

[51] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification," in *Proc. 23rd Int. Conf. Mach. Learn. (ICML)*, 2006, pp. 369–376, doi: 10.1145/1143844.1143891.

[52] O. Abdel-Hamid, A.-R. Mohamed, H. Jiang, and G. Penn, "Applying convolutional neural networks concepts to hybrid NN-HMM model for speech recognition," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Mar. 2012, pp. 4277–4280, doi: 10.1109/ICASSP.2012.6288864.

[53] A. F. R. Nogueira, H. S. Oliveira, J. J. M. Machado, and J. M. R. S. Tavares, "Transformers for urban sound classification—A comprehensive performance evaluation," *Sensors*, vol. 22, no. 22, p. 8874, Nov. 2022, doi: 10.3390/s22228874.

[54] D. M. Ballesteros, Y. Rodriguez-Ortega, D. Renza, and G. Arce, "Deep4SNet: Deep learning for fake speech classification," *Expert Syst. Appl.*, vol. 184, Dec. 2021, Art. no. 115465, doi: 10.1016/j.eswa.2021.115465.

[55] B. Vamsi, M. Mahanty, and B. P. Doppala, "An auto encoder–decoder approach to classify the bird sounds using deep learning techniques," *Social Netw. Comput. Sci.*, vol. 4, no. 3, p. 289, Mar. 2023, doi: 10.1007/s42979-023-01686-4.

[56] P. C. Vakkantula, "Speech mode classification using the fusion of CNNs and LSTM networks," M.S. thesis, Lane Dept. Comput. Sci. Elect. Eng., West Virginia Univ. Libraries, Problem Rep. 7845, 2020, doi: 10.33915/etd.7845.

[57] T. Arias-Vergara, P. Klumpp, J. C. Vasquez-Correa, E. Nöth, J. R. Orozco-Arroyave, and M. Schuster, "Multi-channel spectrograms for speech processing applications using deep learning methods," *Pattern Anal. Appl.*, vol. 24, no. 2, pp. 423–431, Sep. 2020, doi: 10.1007/s10044-020-00921-5.

[58] Y. LeCun and Y. Bengio, "Convolutional networks for images, speech, and time-series," in *The Handbook of Brain Theory and Neural Networks*. Cambridge, MA, USA: MIT Press, 1998, pp. 255–258. [Online]. Available: https://dl.acm.org/doi/10.5555/303568.303704

[59] H.-K. Shin, S. H. Park, and K.-W. Kim, "Inter-floor noise classification using convolutional neural network," *PLoS One*, vol. 15, no. 12, Dec. 2020, Art. no. e0243758, doi: 10.1371/journal.pone.0243758.

[60] H. Choi, H. Yang, S. Lee, and W. Seong, "Classification of inter-floor noise type/position via convolutional neural network-based supervised learning," *Appl. Sci.*, vol. 9, no. 18, p. 3735, Sep. 2019, doi: 10.3390/app9183735.

[61] A. Khamparia, D. Gupta, N. G. Nguyen, A. Khanna, B. Pandey, and P. Tiwari, "Sound classification using convolutional neural network and tensor deep stacking network," *IEEE Access*, vol. 7, pp. 7717–7727, 2019, doi: 10.1109/ACCESS.2018.2888882.

[62] G. Park and S. Lee, "Environmental noise classification using convolutional neural networks with input transform for hearing aids," *Int. J. Environ. Res. Public Health*, vol. 17, no. 7, p. 2270, Mar. 2020, doi: 10.3390/ijerph17072270.

[63] W. Mu, B. Yin, X. Huang, J. Xu, and Z. Du, "Environmental sound classification using temporal-frequency attention based convolutional neural network," *Sci. Rep.*, vol. 11, no. 1, p. 21552, Nov. 2021, doi: 10.1038/s41598-021-01045-4.

[64] Y. A. Al-Hattab, H. F. Zaki, and A. A. Shafie, "Rethinking environmental sound classification using convolutional neural networks: Optimized parameter tuning of single feature extraction," *Neural Comput. Appl.*, vol. 33, no. 21, pp. 14495–14506, May 2021, doi: 10.1007/s00521-021-06091-7.

[65] J. Salamon and J. P. Bello, "Deep convolutional neural networks and data augmentation for environmental sound classification," *IEEE Signal Process. Lett.*, vol. 24, no. 3, pp. 279–283, Mar. 2017, doi: 10.1109/LSP.2017.2657381.

[66] Z. Mushtaq, S.-F. Su, and Q.-V. Tran, "Spectral images based environmental sound classification using CNN with meaningful data augmentation," *Appl. Acoust.*, vol. 172, Jan. 2021, Art. no. 107581, doi: 10.1016/j.apacoust.2020.107581.

[67] S. Hershey, S. Chaudhuri, D. P. W. Ellis, J. F. Gemmeke, A. Jansen, R. C. Moore, M. Plakal, D. Platt, R. A. Saurous, B. Seybold, M. Slaney, R. J. Weiss, and K. Wilson, "CNN architectures for large-scale audio classification," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Mar. 2017, pp. 131–135, doi: 10.1109/ICASSP.2017.7952132.

[68] K. W. Cheng, H. M. Chow, S. Y. Li, T. W. Tsang, H. L. B. Ng, C. H. Hui, Y. H. Lee, K. W. Cheng, S. C. Cheung, C. K. Lee, and S. W. Tsang, "Spectrogram-based classification on vehicles with modified loud exhausts via convolutional neural networks," *Appl. Acoust.*, vol. 205, Mar. 2023, Art. no. 109254, doi: 10.1016/j.apacoust.2023.109254.

[69] M. Dong, "Convolutional neural network achieves human-level accuracy in music genre classification," in *Proc. Conf. Cognit. Comput. Neurosci.*, Feb. 2018, pp. 1–6, doi: 10.32470/ccn.2018.1153-0.

[70] Y. M. G. Costa, L. S. Oliveira, and C. N. Silla, "An evaluation of convolutional neural networks for music classification using spectrograms," *Appl. Soft Comput.*, vol. 52, pp. 28–38, Mar. 2017, doi: 10.1016/j.asoc.2016.12.024.

[71] H. Yang and W.-Q. Zhang, "Music genre classification using duplicated convolutional layers in neural networks," in *Proc. Interspeech*, Sep. 2019, pp. 3382–3386, doi: 10.21437/interspeech.2019-1298.

[72] M. Matocha and S. K. Zieliński, "Music genre recognition using convolutional neural networks," in *Proc. Adv. Comput. Sci. Res.*, 2018, pp. 1–18, doi: 10.24427/ACSR-2018-VOL14-0008.

[73] S. Abdoli, P. Cardinal, and A. L. Koerich, "End-to-end environmental sound classification using a 1D convolutional neural network," *Expert Syst. Appl.*, vol. 136, pp. 252–263, Dec. 2019, doi: 10.1016/j.eswa.2019.06.040.

[74] S. F. Mughal, S. Aamir, S. A. Sahto, and A. Samad, ''Urdu music genre classification using convolution neural networks,'' in *Proc. Int. Conf. Emerg. Trends Smart Technol. (ICETST)*, Sep. 2022, pp. 1–6, doi: 10.1109/ICETST55735.2022.9922934.

[75] W. Suo, ''Efficient music genre classification with deep convolutional neural networks,'' in *Proc. 5th Int. Conf. Data Sci. Inf. Technol. (DSIT)*, Jul. 2022, pp. 01–05, doi: 10.1109/DSIT55514.2022.9943952.

[76] I. Ozer, Z. Ozer, and O. Findik, ''Noise robust sound event classification with convolutional neural network,'' *Neurocomputing*, vol. 272, pp. 505–512, Jan. 2018, doi: 10.1016/j.neucom.2017.07.021.

[77] M. Lesnichaia, V. Mikhailava, N. Bogach, I. Lezhenin, J. Blake, and E. Pyshkin, ''Classification of accented English using CNN model trained on amplitude mel-spectrograms,'' in *Proc. Interspeech*, Sep. 2022, pp. 3669–3673, doi: 10.21437/interspeech.2022-462.

[78] S. S. Malla, ''Acoustic features based accent classification of Kashmiri Language using deep learning,'' *Global J. Comput. Sci. Technol.*, vol. 22, no. 1, pp. 39–43, 2022.

[79] C. Graham, ''L1 identification from L2 speech using neural spectrogram analysis,'' in *Proc. Interspeech*, Aug. 2021, pp. 3959–3963, doi: 10.21437/interspeech.2021-1545.

[80] M. S. Hussain and M. A. Haque, ''SwishNet: A fast convolutional neural network for speech, music and noise classification and segmentation,'' 2018, *arXiv:1812.00149*.

[81] H. Salehghaffari, ''Speaker verification using convolutional neural networks,'' 2018, *arXiv:1803.05427*.

[82] K. Zaman, M. Sah, and C. Direkoglu, ''Classification of harmful noise signals for hearing aid applications using spectrogram images and convolutional neural networks,'' in *Proc. 4th Int. Symp. Multidisciplinary Stud. Innov. Technol. (ISMSIT)*, Oct. 2020, pp. 1–9, doi: 10.1109/ISMSIT50672.2020.9254451.

[83] N. Vrebcevic, I. Mijic, and D. Petrinovic, ''Emotion classification based on convolutional neural network using speech data,'' in *Proc. 42nd Int. Conv. Inf. Commun. Technol., Electron. Microelectron. (MIPRO)*, May 2019, pp. 1007–1012, doi: 10.23919/MIPRO.2019.8756867.

[84] S. Si, J. Wang, H. Sun, J. Wu, C. Zhang, X. Qu, N. Cheng, L. Chen, and J. Xiao, ''Variational information bottleneck for effective low-resource audio classification,'' in *Proc. Interspeech*, Aug. 2021, pp. 591–595, doi: 10.21437/interspeech.2021-2028.

[85] K. Hussain, M. Hussain, and M. S. Khan, ''An improved acoustic scene classification method using convolutional neural networks (CNNs),'' *Amer. Sci. Res. J. Eng., Technol., Sci.*, vol. 44, no. 1, pp. 68–76, Jun. 2018. [Online]. Available: https://asrjetsjournal.org/index.php/American_Scientific_Journal/article/download/4169/1482

[86] A. Dang, T. H. Vu, and J.-C. Wang, ''Acoustic scene classification using convolutional neural networks and multi-scale multi-feature extraction,'' in *Proc. IEEE Int. Conf. Consum. Electron. (ICCE)*, Jan. 2018, pp. 1–4, doi: 10.1109/ICCE.2018.8326315.

[87] T. H. O. Nguyen and F. Pernkopf, ''Acoustic scene classification using a convolutional neural network ensemble and nearest neighbor filters,'' in *Proc. DCASE*, Nov. 2018, pp. 34–38. [Online]. Available: https://graz.pure.elsevier.com/en/publications/acoustic-scene-classification-using-a-convolutional-neural-networ-2

[88] A. Vafeiadis, D. Kalatzis, K. Votis, D, Giakoumis, D. Tzovaras, L. Chen and R. Hamzaoui, ''Acoustic scene classification: From a hybrid classifier to deep learning,'' in *Proc. Detection Classification Acoustic Scenes Events*, Nov. 2017. [Online]. Available: https://dcase.community/documents/challenge2017/technical_reports/DCASE2017_Vafeiadis_134.pdf

[89] L. Pham, I. McLoughlin, H. Phan, and R. Palaniappan, ''A robust framework for acoustic scene classification,'' in *Proc. Interspeech*, Sep. 2019, pp. 3634–3638, doi: 10.21437/interspeech.2019-1841.

[90] K. K. Jena, S. K. Bhoi, S. Mohapatra, and S. Bakshi, ''A hybrid deep learning approach for classification of music genres using wavelet and spectrogram analysis,'' *Neural Comput. Appl.*, vol. 35, no. 15, pp. 11223–11248, Jan. 2023, doi: 10.1007/s00521-023-08294-6.

[91] M. Scarpiniti, D. Comminiello, A. Uncini, and Y.-C. Lee, ''Deep recurrent neural networks for audio classification in construction sites,'' in *Proc. 28th Eur. Signal Process. Conf. (EUSIPCO)*, Jan. 2021, pp. 810–814, doi: 10.23919/EUSIPCO47968.2020.9287802.

[92] J. Sang, S. Park, and J. Lee, ''Convolutional recurrent neural networks for urban sound classification using raw waveforms,'' in *Proc. 26th Eur. Signal Process. Conf. (EUSIPCO)*, Sep. 2018, pp. 2444–2448, doi: 10.23919/EUSIPCO.2018.8553247.

[93] K. Liao, Y. Zhao, J. Gu, Y. Zhang, and Y. Zhong, ''Sequential convolutional recurrent neural networks for fast automatic modulation classification,'' *IEEE Access*, vol. 9, pp. 27182–27188, 2021, doi: 10.1109/ACCESS.2021.3053427.

[94] N. Asatani, T. Kamiya, S. Mabu, and S. Kido, ''Classification of respiratory sounds using improved convolutional recurrent neural network,'' *Comput. Electr. Eng.*, vol. 94, Sep. 2021, Art. no. 107367, doi: 10.1016/j.compeleceng.2021.107367.

[95] Z. Zhang, S. Xu, S. Zhang, T. Qiao, and S. Cao, ''Attention based convolutional recurrent neural network for environmental sound classification,'' *Neurocomputing*, vol. 453, pp. 896–903, Sep. 2021, doi: 10.1016/j.neucom.2020.08.069.

[96] K. Choi, G. Fazekas, M. Sandler, and K. Cho, ''Convolutional recurrent neural networks for music classification,'' in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Mar. 2017, pp. 2392–2396, doi: 10.1109/ICASSP.2017.7952585.

[97] C. Federico, B. Federica, N. Alessandro, and C. Marco, ''Convolutional recurrent neural network for audio events classification detection and classification of acoustic scenes and events 2018,'' Tech. Rep., 2018. [Online]. Available: https://dcase.community/documents/challenge2018/technical_reports/DCASE2018_Colangelo_61.pdf

[98] Z. Nasrullah and Y. Zhao, ''Music artist classification with convolutional recurrent neural networks,'' 2019, *arXiv:1901.04555*.

[99] L. Feng, S. Liu, and J. Yao, ''Music genre classification with paralleling recurrent convolutional neural network,'' 2017, *arXiv:1712.08370*.

[100] Y. Yu, S. Luo, S. Liu, H. Qiao, Y. Liu, and L. Feng, ''Deep attention based music genre classification,'' *Neurocomputing*, vol. 372, pp. 84–91, Jan. 2020, doi: 10.1016/j.neucom.2019.09.054.

[101] J. Gan, ''Music feature classification based on recurrent neural networks with channel attention mechanism,'' *Mobile Inf. Syst.*, vol. 2021, Jun. 2021, Art. no. 7629994, doi: 10.1155/2021/7629994.

[102] N. Srivastava, S. Ruhil, and G. Kaushal, ''Music genre classification using convolutional recurrent neural networks,'' in *Proc. IEEE 6th Conf. Inf. Commun. Technol. (CICT)*, Gwalior, India, Nov. 2022, pp. 1–5, doi: 10.1109/CICT56698.2022.9997961.

[103] M. Nigro, A. Rueda, and S. Krishnan, ''Acoustic scene classification using time-frequency energy emphasis and convolutional recurrent neural networks,'' in *Artificial Intelligence and Evolutionary Computations in Engineering Systems*. Cham, Switzerland: Springer, 2021, pp. 267–276, doi: 10.1007/978-981-16-2674-6_21.

[104] T. Qiao, S. Zhang, Z. Zhang, S. Cao, and S. Xu, ''Sub-spectrogram segmentation for environmental sound classification via convolutional recurrent neural network and score level fusion,'' in *Proc. IEEE Int. Workshop Signal Process. Syst. (SiPS)*, Oct. 2019, pp. 318–323, doi: 10.1109/SIPS47522.2019.9020418.

[105] H. Allet, E. Cakır, and T. Virtanen, ''Acoustic scene classification using convolutional recurrent neural networks,'' in *Proc. Detection Classification Acoustic Scenes Events (DCASE)*, 2017, pp. 1–5. [Online]. Available: https://dcase.community/documents/challenge2017/technical_reports/DCASE2017_Jallet_140.pdf

[106] S. Adavanne, P. Pertilä, and T. Virtanen, ''Sound event detection using spatial features and convolutional recurrent neural network,'' in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Mar. 2017, pp. 771–775, doi: 10.1109/ICASSP.2017.7952260.

[107] R. Yang, L. Feng, H. Wang, J. Yao, and S. Luo, ''Parallel recurrent convolutional neural networks-based music genre classification method for mobile devices,'' *IEEE Access*, vol. 8, pp. 19629–19637, 2020, doi: 10.1109/ACCESS.2020.2968170.

[108] M. Shah, N. Pujara, K. Mangaroliya, L. Gohil, T. Vyas, and S. Degadwala, ''Music genre classification using deep learning,'' in *Proc. 6th Int. Conf. Comput. Methodologies Commun. (ICCMC)*, Mar. 2022, pp. 974–978, doi: 10.1109/ICCMC53470.2022.9753953.

[109] G. Gupta, M. Kshirsagar, M. Zhong, S. Gholami, and J. L. Ferres, ''Comparing recurrent convolutional neural networks for large scale bird species classification,'' *Sci. Rep.*, vol. 11, no. 1, p. 17085, Aug. 2021, doi: 10.1038/s41598-021-96446-w.

[110] K. Zhang, ''Music style classification algorithm based on music feature extraction and deep neural network,'' *Wireless Commun. Mobile Comput.*, vol. 2021, Sep. 2021, Art. no. 9298654, doi: 10.1155/2021/9298654.

[111] K. Banuroopa and D. S. Priyaa, ''MFCC based hybrid fingerprinting method for audio classification through LSTM,'' *Int. J. Nonlinear Anal. Appl.*, vol. 12, pp. 2125–2136, Dec. 2021, doi: 10.22075/ijnaa.2022.6049.

[112] V. A. Lakhani and R. Mahadev, ''Multi-language identification using convolutional recurrent neural network,'' 2016, *arXiv:1611.04010*.

[113] M. H. Soni and H. A. Patil, "Novel deep autoencoder features for non-intrusive speech quality assessment," in *Proc. 24th Eur. Signal Process. Conf. (EUSIPCO)*, Budapest, Hungary, Aug. 2016, pp. 2315–2319, doi: 10.1109/EUSIPCO.2016.7760662.

[114] J. Naranjo-Alcazar, S. Perez-Castanos, P. Zuccarello, F. Antonacci, and M. Cobos, "Open set audio classification using autoencoders trained on few data," *Sensors*, vol. 20, no. 13, p. 3741, Jul. 2020, doi: 10.3390/s20133741.

[115] S. Amiriparian, M. Freitag, N. Cummins, and B. Schuller, "Sequence to sequence autoencoders for unsupervised representation learning from audio," in *Proc. DCASE*, 2017, pp. 17–21.

[116] P.-Y. Huang, H. Xu, J. Li, A. Baevski, M. Auli, W. Galuba, F. Metze, and C. Feichtenhofer, "Masked autoencoders that listen," 2022, *arXiv:2207.06405*.

[117] J. Abeßer, S. I. Mimilakis, R. Gräfe, H. M. Lukashevich, and I. D. M. T. Fraunhofer, "Acoustic scene classification by combining autoencoder-based dimensionality reduction and convolutional neural networks," in *Proc. DCASE*, Nov. 2017, pp. 7–11. [Online]. Available: https://publica.fraunhofer.de/entities/publication/5c3af26b-bc82-4911-9553-d7099f00b385/details

[118] Y. Atahan, A. Elbir, A. E. Keskin, O. Kiraz, B. Kirval, and N. Aydin, "Music genre classification using acoustic features and autoencoders," in *Proc. Innov. Intell. Syst. Appl. Conf. (ASYU)*, Oct. 2021, pp. 1–5, doi: 10.1109/ASYU52992.2021.9598979.

[119] J. Qi and H. Van Hamme, "Speech disorder classification using extended factorized hierarchical variational auto-encoders," in *Proc. Interspeech*, Aug. 2021, pp. 1–5, doi: 10.21437/interspeech.2021-2180.

[120] Z. Lin, Y. Li, Z. Huang, W. Zhang, Y. Tan, Y. Chen, and Q. He, "Domestic activities clustering from audio recordings using convolutional capsule autoencoder network," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Jun. 2021, pp. 835–839, doi: 10.1109/ICASSP39728.2021.9414643.

[121] A. Sawhney, V. Vasavada, and W. Wang, "Latent feature extraction for musical genres from raw audio," in *Proc. 32nd Conf. Neural Inf. Process. Syst. (NIPS)*, Montréal, QC, Canada, Dec. 2021, pp. 2–8. [Online]. Available: https://cs229.stanford.edu/proj2018/report/20.pdf

[122] L. Qiu, S. Li, and Y. Sung, "3D-DCDAE: Unsupervised music latent representations learning method based on a deep 3D convolutional denoising autoencoder for music genre classification," *Mathematics*, vol. 9, no. 18, p. 2274, Sep. 2021, doi: 10.3390/math9182274.

[123] J. Saldanha, S. Chakraborty, S. Patil, K. Kotecha, S. Kumar, and A. Nayyar, "Data augmentation using Variational Autoencoders for improvement of respiratory disease classification," *PLoS One*, vol. 17, no. 8, Aug. 2022, Art. no. e0266467, doi: 10.1371/journal.pone.0266467.

[124] S. Latif, R. Rana, J. Qadir, and J. Epps, "Variational Autoencoders for learning latent representations of speech emotion: A preliminary study," in *Proc. Interspeech*, Sep. 2018, pp. 3107–3111, doi: 10.21437/interspeech.2018-1568.

[125] M. Asni, D. Shapiro, M. Bolic, T. Mathew, and L. Grebler, "Speaker differentiation using a convolutional autoencoder," in *Proc. IEEE Int. Conf. Sci. Electr. Eng. Isr. (ICSEE)*, Dec. 2018, pp. 1–5, doi: 10.1109/ICSEE.2018.8646307.

[126] K. Wilkinghoff and F. Kurth, "Open-set acoustic scene classification with deep convolutional autoencoders," in *Proc. Detection Classification Acoustic Scenes Events Workshop (DCASE)*, Oct. 2019, pp. 258–262, doi: 10.33682/340j-wd27.

[127] M. Irfan, Z. Jiangbin, S. Ali, M. Iqbal, Z. Masood, and U. Hamid, "DeepShip: An underwater acoustic benchmark dataset and a separable convolution based autoencoder for classification," *Expert Syst. Appl.*, vol. 183, Nov. 2021, Art. no. 115270, doi: 10.1016/j.eswa.2021.115270.

[128] S. Arniriparian, M. Freitag, N. Cummins, M. Gerczuk, S. Pugachevskiy, and B. Schuller, "A fusion of deep convolutional generative adversarial networks and sequence to sequence autoencoders for acoustic scene classification," in *Proc. 26th Eur. Signal Process. Conf. (EUSIPCO)*, Sep. 2018, pp. 977–981, doi: 10.23919/EUSIPCO.2018.8553225.

[129] M. Freitag, S. Amiriparian, S. Pugachevskiy, N. Cummins, and B. Schuller, "auDeep: Unsupervised learning of representations from audio with deep recurrent neural networks," *J. Mach. Learn. Res.*, vol. 18, no. 173, pp. 1–5, Jan. 2018. [Online]. Available: https://jmlr.org/papers/volume18/17-406/17-406.pdf

[130] A. Czyżewski, A. Kurowski, and S. Zaporowski, "Application of autoencoder to traffic noise analysis," in *Proc. Meetings Acoust.*, 2019, Art. no. 055003, doi: 10.1121/2.0001227.

[131] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," 2017, *arXiv:1706.03762*.

[132] Y. Gong, Y.-A. Chung, and J. Glass, "AST: Audio spectrogram transformer," 2021, *arXiv:2104.01778*.

[133] K. Koutini, J. Schlüter, H. Eghbal-zadeh, and G. Widmer, "Efficient training of audio transformers with patchout," in *Proc. Interspeech*, Sep. 2022, pp. 2753–2757, doi: 10.21437/interspeech.2022-227.

[134] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16 × 16 words: Transformers for image recognition at scale," 2020, *arXiv:2010.11929*.

[135] Y. Zhang, B. Li, H. Fang, and Q. Meng, "Spectrogram transformers for audio classification," in *Proc. IEEE Int. Conf. Imag. Syst. Techn. (IST)*, Jun. 2022, pp. 1–6, doi: 10.1109/IST55454.2022.9827729.

[136] J. Luo, J. Yang, E. S. Chng, and X. Zhong, "Vision transformer based audio classification using patch-level feature fusion," in *Proc. Asia–Pacific Signal Inf. Process. Assoc. Annu. Summit Conf. (APSIPA ASC)*, Nov. 2022, pp. 22–26, doi: 10.23919/APSIPAASC55919.2022.9980194.

[137] Y. Gong, Y.-A. Chung, and J. Glass, "AST: Audio Spectrogram Transformer," in *Proc. Interspeech*, Aug. 2021, pp. 1–5, doi: 10.21437/interspeech.2021-698.

[138] S. Wyatt, D. Elliott, A. Aravamudan, C. E. Otero, L. D. Otero, G. C. Anagnostopoulos, A. O. Smith, A. M. Peter, W. Jones, S. Leung, and E. Lam, "Environmental sound classification with tiny transformers in noisy edge environments," in *Proc. IEEE 7th World Forum Internet Things (WF-IoT)*, Jun. 2021, pp. 309–314, doi: 10.1109/WF-IoT51360.2021.9596007.

[139] Y. Gong, C.-I. Lai, Y.-A. Chung, and J. Glass, "SSAST: Self-supervised audio spectrogram transformer," in *Proc. AAAI Conf. Artif. Intell.*, Jun. 2022, vol. 36, no. 10, pp. 10699–10709, doi: 10.1609/aaai.v36i10.21315.

[140] K. Koutini, J. Schlüter, H. Eghbal-zadeh, and G. Widmer, "Efficient training of audio transformers with patchout," 2021, *arXiv:2110.05069*.

[141] H. Zhao, C. Zhang, B. Zhu, Z. Ma, and K. Zhang, "S3T: Self-supervised pre-training with Swin transformer for music classification," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2022, pp. 606–610, doi: 10.1109/ICASSP43922.2022.9746056.

[142] S. Atito, M. Awais, W. Wang, M. D. Plumbley, and J. Kittler, "ASiT: Audio spectrogram vision transformer for general audio representation," 2022, *arXiv:2211.13189*.

[143] X. Liu, H. Lu, J. Yuan, and X. Li, "CAT: Causal audio transformer for audio classification," 2023, *arXiv:2303.07626*.

[144] J. Zeng, D. Zhang, Z. Li, and X. Li, "Semi-supervised training of transformer and causal dilated convolution network with applications to speech topic classification," *Appl. Sci.*, vol. 11, no. 12, p. 5712, Jun. 2021, doi: 10.3390/app11125712.

[145] Y. Zhuang, Y. Chen, and J. Zheng, "Music Genre classification with transformer classifier," in *Proc. 4th Int. Conf. Digit. Signal Process. (ACM)*, Jun. 2020, pp. 155–159, doi: 10.1145/3408127.3408137.

[146] L. Qiu, S. Li, and Y. Sung, "DBTMPE: Deep bidirectional transformers-based masked predictive encoder approach for music genre classification," *Mathematics*, vol. 9, no. 5, p. 530, Mar. 2021, doi: 10.3390/math9050530.

[147] P. Verma and J. Berger, "Audio transformers:Transformer architectures for large scale audio understanding. Adieu convolutions," 2021, *arXiv:2105.00335*.

[148] K. Chen, X. Du, B. Zhu, Z. Ma, T. Berg-Kirkpatrick, and S. Dubnov, "HTS-AT: A hierarchical token-semantic audio transformer for sound classification and detection," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2022, pp. 646–650, doi: 10.1109/ICASSP43922.2022.9746312.

[149] P. Primus and G. Widmer, "Improved zero-shot audio tagging & classification with patchout spectrogram transformers," in *Proc. 30th Eur. Signal Process. Conf. (EUSIPCO)*, Aug. 2022, pp. 410–413, doi: 10.23919/eusipco55093.2022.9909760.

[150] S. Ghosh, A. Seth, S. Umesh, and D. Manocha, "MAST: Multiscale audio spectrogram transformers," 2022, *arXiv:2211.01515*.

[151] W. Zhu and M. Omar, "Multiscale audio spectrogram transformer for efficient audio classification," 2023, *arXiv:2303.10757*.

[152] H. Akbari, L. Yuan, R. Qian, W.-H. Chuang, S.-F. Chang, Y. Cui, and B. Gong, "VATT: Transformers for multimodal self-supervised learning from raw video, audio and text," 2021, *arXiv:2104.11178*.

[153] F. Demir, M. Turkoglu, M. Aslan, and A. Sengur, "A new pyramidal concatenated CNN approach for environmental sound classification," *Appl. Acoust.*, vol. 170, Dec. 2020, Art. no. 107520, doi: 10.1016/j.apacoust.2020.107520.

[154] H. Phan, P. Koch, F. Katzberg, M. Maass, R. Mazur, and A. Mertins, "Audio scene classification with deep recurrent neural networks," 2017, *arXiv:1703.04770*.

[155] H. Li, S. Xue, and J. Zhang. (2018). *Combining CNN and Classical Algorithms for Music Genre Classification*. [Online]. Available: https://cs229.stanford.edu/proj2018/report/19.pdf

[156] A. H. Falah and J. Jondri, "Lung sounds classification using stacked autoencoder and support vector machine," in *Proc. 7th Int. Conf. Inf. Commun. Technol. (ICoICT)*, Jul. 2019, pp. 1–5, doi: 10.1109/ICOICT.2019.8835278.

[157] S. L. Ullo, S. K. Khare, V. Bajaj, and G. R. Sinha, "Hybrid computerized method for environmental sound classification," *IEEE Access*, vol. 8, pp. 124055–124065, 2020, doi: 10.1109/ACCESS.2020.3006082.

[158] H. Tang and N. Chen, "Combining CNN and broad learning for music classification," *IEICE Trans. Inf. Syst.*, vol. 103, no. 3, pp. 695–701, Mar. 2020, doi: 10.1587/transinf.2019edp7175.

[159] P. Fulzele, R. Singh, N. Kaushik, and K. Pandey, "A hybrid model for music genre classification using LSTM and SVM," in *Proc. 11th Int. Conf. Contemp. Comput. (IC)*, Aug. 2018, pp. 1–3, doi: 10.1109/IC3.2018.8530557.

[160] M. Sah and C. Direkoglu, "A survey of deep learning methods for multiple sclerosis identification using brain MRI images," *Neural Comput. Appl.*, vol. 34, no. 10, pp. 7349–7373, Mar. 2022, doi: 10.1007/s00521-022-07099-3.

[161] B. Zhang, J. Leitner, and S. Thornton. (2019). *Audio Recognition Using Mel Spectrograms and Convolution Neural Networks*. [Online]. Available: http://noiselab.ucsd.edu/ECE228_2019/Reports/Report38.pdf

[162] M. Bandara, R. Jayasundara, I. Ariyarathne, D. Meedeniya, and C. Perera, "Forest sound classification dataset: FSC22," *Sensors*, vol. 23, no. 4, p. 2032, Feb. 2023, doi: 10.3390/s23042032.

[163] K. J. Piczak, "ESC: Dataset for environmental sound classification," in *Proc. 23rd ACM Int. Conf. Multimedia. (ACM)*, Oct. 2015, pp. 1015–1018, doi: 10.1145/2733373.2806390.

[164] J. Salamon, C. Jacoby, and J. P. Bello, "A dataset and taxonomy for urban sound research," in *Proc. 22nd ACM Int. Conf. Multimedia*, Jul. 2014, pp. 1041–1044, doi: 10.1145/2647868.2655045.

[165] B. L. Sturm, "An analysis of the GTZAN music genre dataset," in *Proc. 2nd Int. ACM workshop Music Inf. Retr. User-Centered Multimodal Strategies*, Nov. 2012, pp. 7–12, doi: 10.1145/2390848.2390851.

[166] Mozilla. *Mozilla Common Voice Dataset*. Accessed: Jun. 7, 2022. [Online]. Available: https://commonvoice.mozilla.org/en

[167] A. Nagrani, J. S. Chung, and A. Zisserman, "VoxCeleb: A large-scale speaker identification dataset," in *Proc. Interspeech*, Aug. 2017, pp. 2616–2620, doi: 10.21437/interspeech.2017-950.

[168] F. Burkhardt, A. Paeschke, M. Rolfes, W. F. Sendlmeier, and B. Weiss, "A database of German emotional speech," in *Proc. Interspeech*, Sep. 2005, pp. 1517–1520, doi: 10.21437/interspeech.2005-446.

[169] D. Snyder, G. Chen, and D. Povey, "MUSAN: A music, speech, and noise corpus," 2015, *arXiv:1510.08484*.

[170] B. M. Rocha, D. Filos, L. Mendes, G. Serbes, S. Ulukaya, Y. P. Kahya, N. Jakovljevic, T. L. Turukalo, I. M. Vogiatzis, E. Perantoni, E. Kaimakamis, P. Natsiavas, A. Oliveira, C. Jácome, A. Marques, N. Maglaveras, R. P. Paiva, I. Chouvarda, and P. de Carvalho, "An open access database for the evaluation of respiratory sound classification algorithms," *Physiolog. Meas.*, vol. 40, no. 3, Mar. 2019, Art. no. 035001, doi: 10.1088/1361-6579/ab03ea.

[171] A. Mesaros, T. Heittola, and T. Virtanen, Mar. 17, 2017, "TUT Sound Events 2017, Development Dataset," Zenodo, doi: 10.5281/ZENODO.400516.

[172] A. Mesaros, T. Heittola, and T. Virtanen, "TUT database for acoustic scene classification and sound event detection," in *Proc. 24th Eur. Signal Process. Conf. (EUSIPCO)*, Aug. 2016, pp. 1128–1132, doi: 10.1109/EUSIPCO.2016.7760424.

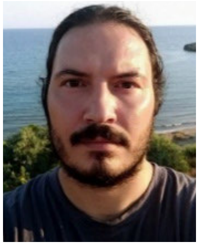
[173] P. Warden, "Speech commands: A dataset for limited-vocabulary speech recognition," 2018, *arXiv:1804.03209*.

[174] T. Heittola. (2017). *DCASE2017 Challenge—DCASE*. [Online]. Available: https://dcase.community/challenge2017/

[175] N. Lopac, F. Hržić, I. P. Vuksanovic, and J. Lerga, "Detection of non-stationary GW signals in high noise from Cohen's class of time-frequency representations using deep learning," *IEEE Access*, vol. 10, pp. 2408–2428, 2022, doi: 10.1109/ACCESS.2021.3139850.

KHALID ZAMAN received the bachelor's degree in telecommunication and networks from Abasyn University, Peshawar, Pakistan, in 2016, and the master's degree in computer engineering from Near East University, North Cyprus, Turkey, in 2019. He is currently pursuing the Ph.D. degree with the Graduate School of Advanced Science and Technology, Japan Advanced Institute of Science and Technology (JAIST), Japan. His current research interests include digital speech processing and deepfake audio classification, with a focus on applying deep learning methods.

MELIKE SAH received the B.Sc. and M.Sc. degrees in computer engineering from Eastern Mediterranean University, North Cyprus, and the Ph.D. degree in computer science from the University of Southampton, U.K., in 2009. She was a Postdoctoral Researcher with Trinity College Dublin, Ireland, from 2009 to 2014. She is currently a Professor with Cyprus International University, North Cyprus. Her current research interests include artificial intelligence, deep learning, biomedical signal/data analysis, semantic web/web-based systems, search/retrieval models, and computer vision. Currently, she is an Associate Editor of IEEE Transactions on Big Data and IEEE Transactions on Artificial Intelligence.



CEM DIREKOGLU received the B.Sc. and M.Sc. degrees in electrical and electronics engineering from Eastern Mediterranean University and the Ph.D. degree in computer vision from the University of Southampton. He was a Postdoctoral Researcher with the Graphics, Vision and Visualization Group, School of Computer Science and Statistics, Trinity College Dublin. From December 2010 to August 2014, he was a Postdoctoral Researcher with the INSIGHT Centre for Data Analytics, Dublin City University. Since September 2014, he has been a full-time Faculty Member with the Electrical and Electronics Engineering Program, Middle East Technical University, Northern Cyprus Campus. His current research interests include computer vision, signal processing, and deep learning. He was a member of the Information Signals, Images and Systems (ISIS) Research Group, School of Electronics and Computer Science.

MASASHI UNOKI (Member, IEEE) received the M.S. and Ph.D. degrees in information science from the Japan Advanced Institute of Science and Technology (JAIST), in 1996 and 1999, respectively. He was a Japan Society for the Promotion of Science (JSPS) Research Fellow, from 1998 to 2001. He was a Visiting Researcher with the ATR Human Information Processing Laboratories, from 1999 to 2000, and a Visiting Research Associate with the Centre for the Neural Basis of Hearing (CNBH), Department of Physiology, University of Cambridge, from 2000 to 2001. He has been a Faculty Member with the School of Information Science, JAIST, since 2001, where he is currently a Full Professor and the Dean of the School of Information Science. His current research interests include auditory-motivated signal processing and the modeling of auditory systems. He is an IEICE Fellow. Currently, he is an Associate Editor of *Applied Acoustics*.

• • •